Mining, Analyzing and Exploiting Community Feedback on the Web

Von der Fakultät für Elektrotechnik und Informatik der
Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades
Doktor der Naturwissenschaften
Dr. rer. nat.

genehmigte Dissertation

von

Dipl.-Ing. Sergiu Chelaru
geboren am 1984/02/11 in Iasi, Rumänien

Referent: Prof. Dr. techn. Wolfgang Nejdl
Koreferent: Asst. Prof. Dr. Ismail Sengör Altingövde
Koreferent: Prof. Dr. Heribert Vollmer

Tag der Promotion: 2014/10/13

Zusammenfassung

Moderner Web-Plattformen, welche in den vergangenen Jahren stark an Popularität gewonnen haben, stellen verschiedenste Möglichkeiten zur Interaktion und Kommunikation bereit. Portale wie YouTube und Yahoo! News etwa erlauben es Nutzern die veröffentlichten Inhalte zu kommentieren sowie auf Kommentare andere Nutzer zu antworten und diese zu bewerten. Diese Art des expliziten Community-Feedbacks stellt eine interessante Quelle für weitere Erkenntnisse dar: Die Analyse dieser Daten erlaubt es implizites Wissen über das Teilen von Medien und Interessen von Nutzern und Communities zu gewinnen.

Der erste Teil dieser Arbeit ist eine detaillierte Studie des Kommentar-basierten Feedbacks, basierend auf Datensätzen globaler Inhaltsanbieter. Hierbei werden verschiedene Arten von Social Media Anwendungen und damit verschiedene Arten von Feedback untersucht. Weiterhin wird die Anwendbarkeit von Methoden des maschinellen Lernens sowie Data-Mining-Techniken analysiert. Ziel ist es hierbei, Vorhersagen darüber zu treffen, ob Kommentare akzeptiert werden, ob sie Diskussionen auslösen, ob sie kontrovers sind und ob sie offensives Verhalten von Nutzern auslösen.

Zahlreiche Web 2.0 Plattformen bieten Nutzern zusätzliche Möglichkeiten zur Interaktion mit geteilten Inhalten wie zum Beispiel die Markierung mit “likes”, “dislikes” oder “favourites”. Diese Interaktionen generieren eine große Menge an Daten zu sozialem Feedback. Hierbei stellt sich die Frage, ob sich aus diesem sozialen Feedback Eigenschaften extrahieren lassen, welche es erlauben relevantere Inhalte oder auch Inhalte von höherer Qualität zu finden. Trotz des wachsenden Interesses an Web 2.0 Anwendungen, sowohl von Seiten der Industrie als auch von Forschern verschiedener Disziplinen, ist diese Frage noch nicht vollständig beantwortet. Im zweiten Teil dieser Arbeit wird daher der Einfluss dieser sozialen Features auf die Suche nach Videos in YouTube untersucht. Hierbei werden dem Stand der Technik entsprechende Learning-to-Rank Methoden angewandt. Die durchgeführten Experimente zeigen, dass soziale Features vielversprechende Ergebnisse liefern und die Suchergebnisse für Videos in YouTube verbessern können.

Abschließend widmet sich diese Arbeit einer anderen Art des impliziten Feedbacks im Web, den in Suchanfragen ausgedrückten Meinungen und Stimmungen von Communities. Ziel ist es hierbei Suchanfragen als eine neue und wenig genutzte Quelle von benutzergenerierten Inhalten zu verwenden. Hierdurch sollen Ansichten und Meinungen zu kontroversen Themen aufgedeckt werden. Unseres Wissens nach ist dies die erste Arbeit, die eine detaillierte Charakterisierung von Suchanfragen in Bezug auf die in ihnen ausgedrückten Meinungen erstellt. Darüber hinaus wurden verschiedene Modelle entwickelt und evaluiert, um die Stimmung in einer Anfrage vorherzusagen. Außerdem untersucht die Arbeit den Nutzen dieser Vorhersagen für die Erkennung kontroverser Themen und die Empfehlung von Suchbegriffen.

Schlagwörter: *Community Feedback, Comment Ratings, Social Features, Learning to Rank, Sentiment Analysis, Opinionated Queries*

Abstract

In recent years we have witnessed an increasing number of modern Web platforms that provide various tools for community interaction. For instance, YouTube and Yahoo! News include the mechanism to comment on the published content and users are able to rate and reply to comments. This type of explicit community feedback constitutes a potentially interesting data source to mine for obtaining implicit knowledge about shared media, users and community interests.

In this thesis, we first conduct an in-depth study of *comment-centric feedback* on real world datasets crawled from top content providers, covering different types of social media with different underlying feedback behavior. Furthermore, we explore the applicability of machine learning and data mining techniques to predict the acceptance of comments, comments likely to trigger discussions, controversial comments, and users exhibiting offensive commenting behavior.

Numerous Web 2.0 platforms offer other additional ways for users to interact with the shared content (e.g., "likes", "dislikes","favorites"), resulting in a huge amount of *social feedback*. Can the features extracted from the social feedback help the underlying search systems for guiding its users to reach to a better quality or more relevant content? Despite the rapid and growing interest for Web 2.0 applications from both the industry and researchers from various disciplines, this question is still not clearly answered. In the second part of this thesis, we investigate the impact of social features on the effectiveness of video retrieval in YouTube using state-of-the-art learning to rank techniques. Our experiments reveal that social features are promising and can improve the retrieval performance for videos in YouTube.

Finally, we refer to another type of implicit feedback on the web, namely *community sentiment* in Web queries. Our objective here is to analyze and exploit Web queries as a new, rich and mostly unexplored source of user-generated content that can convey community views and opinions on a multitude of controversial topics. To the best of our knowledge, we are the first to provide a detailed characterization of search queries in terms of opinions expressed in them. Furthermore, we build and evaluate different models to predict the sentiment in a query. Finally, we demonstrate the virtue of query sentiment detection for the tasks of controversial topic discovery and query recommendation.

Keywords: *Community Feedback, Comment Ratings, Social Features, Learning to Rank, Sentiment Analysis, Opinionated Queries*

Contents

List of Figures **vi**

List of Tables **x**

1 Introduction **1**

1.1 Motivation 1

1.2 Overview of Contributions 4

1.3 Outline of the Thesis 7

2 Background **9**

2.1 Text Classification 9

2.2 Sentiment Analysis 11

2.3 Social Content and Features 13

2.4 Learning to Rank (LETOR) 14

3 Comment-Centric Feedback on the Social Web **19**

3.1 Related Work 19

3.2 Data Gathering, Methods and Characteristics 21

3.2.1 Data Gathering 21

3.2.2 Data Characteristics 23

3.3 Comment Ratings 25

3.3.1 Term Analysis of Rated Comments 25

3.3.2 Sentiment Analysis of Rated Comments 28

3.3.3 Temporal Characteristics of Comment Ratings 30

3.3.4 Comment Ratings and Polarizing Content 32

3.3.5 Predicting Comment Ratings . . . 35
3.3.6 Category-Specific Rating Prediction . . . 38
3.4 Discussion Threads and Replies . . . 40
3.4.1 Analysis of Replies and Ratings . . . 40
3.4.2 Predicting the Responsivity on Comments . . . 43
3.5 Controversial Comments . . . 45
3.5.1 Analysis of "likes" and "dislikes" . . . 45
3.5.2 Term Analysis of Controversial Comments . . . 48
3.5.3 Analysis of Ratings for Comment Threads . . . 49
3.5.4 Predicting Controversial Comments . . . 50
3.6 Users Commenting on Social Web Environments . . . 51
3.6.1 Finding Trolls . . . 51
3.6.2 Trolls and Community Ratings . . . 52
3.6.3 Content-based Troll Prediction . . . 53
3.7 Summary and Contributions . . . 56

4 Social Feedback **57**
4.1 Related Work . . . 57
4.2 Data Gathering, Methods and Characteristics . . . 58
4.2.1 Query Sets . . . 58
4.2.2 Query Volume Analysis . . . 60
4.2.3 Statistics for the Result Videos . . . 63
4.2.4 Query Result Characterization using Social Features . . . 64
4.3 Effectiveness and Correlation of Individual Features . . . 67
4.4 Learning to Rank using Social Features . . . 73
4.4.1 Video Retrieval Framework . . . 75
4.4.2 Experimental Results for Feature Selection . . . 76
4.4.3 Experimental Results for the Impact of Social Features . . . 77
4.5 Summary and Contributions . . . 81

5 Community Sentiment in Web Queries **83**
5.1 Related Work . . . 83
5.2 Data Gathering, Methods and Characteristics . . . 85
5.2.1 Data Collection . . . 85
5.2.2 Characteristics of Opinionated Queries . . . 87
5.2.3 Sentiment in Web Search Queries . . . 87
5.2.4 Analysis of Query Volumes . . . 90
5.2.5 Sentiment in Query Results . . . 91

5.2.6 Post-Retrieval Analysis 92
5.2.7 Lexicon-Based Sentiment Analysis 94
5.2.8 Regional Analysis 96
5.3 Detecting Query Sentiment 99
5.4 Application Scenarios 102
5.4.1 Query Recommendation 102
5.4.2 Controversial Topic Discovery 105
5.5 Summary and Contributions 109

6 Conclusions and Future Work **111**

Bibliography **114**

Curriculum Vitae **127**

List of Figures

1.1 Examples of comments and ratings in YouTube. 2
1.2 Top-3 videos for the query "michelle phan" re-ranked using the query-title similarity (upper row) and comment ratings (lower row). 3

2.1 Maximum margin decision hyperplane for a linear SVM. 10
2.2 Examples of positive (top) and negative (bottom) opinionated movie reviews (from Amazon). 12
2.3 The Opinion Mining and Aggregation System developed in [45]. 13
2.4 A Learning to Rank framework. 15

3.1 Distribution of number of comments for videos in YouTube and news stories in Yahoo! News. 24
3.2 Distribution of comment ratings for (a) YouTube, and (b) Yahoo! News. 25
3.3 Distribution of comment negativity, and positivity for (a) YouTube and (b) Yahoo! News. 29
3.4 Comparison of mean senti-values for comments with different kinds of community ratings in (a) YouTube and (b) Yahoo! News. 30
3.5 Crawling strategy for temporal analysis. 31
3.6 Temporal evolution of average number of comment ratings, likes, and dislikes; and average number of replies. 31
3.7 Temporal evolution of average acceptance ratios for different ranges of values for initial comment acceptance ratios Φ. 32
3.8 Videos with high (upper row) versus low variance (lower row) of comment ratings. 33
3.9 Precision-recall curves for comment rating prediction. 37

3.10 Precision-recall curves for category-specific rating prediction experiments. 39
3.11 Distribution of thread sizes (number of replies) for the YouTube and Yahoo! News corpora. 41
3.12 Distribution of ratings for seed and reply comments in (a) the YouTube dataset and (b) the Yahoo! News dataset. 42
3.13 Average comment rating of seed comments in Yahoo! News and YouTube with respect to thread size. 43
3.14 Precision-recall curves for predicting replied comments for (a) the YouTube and (b) the Yahoo! News dataset. 44
3.15 (a) Distribution of number of comments per comment approval (Φ) intervals for distinct thresholds θ for the number of received ratings. (b) Controversy interval vs. accepted (positive) and not accepted (negative) intervals. 47
3.16 Distribution of controversial comments for distinct thresholds θ for the number of received comment ratings. 48
3.17 Comment ratings and controversy with respect to thread size. 49
3.18 Precision-recall and ROC curve for the classification of controversial comments (δ_{NC}=0.4). 51
3.19 Distribution of troll and non-troll users in YouTube with respect to user approval ratio (Ψ) intervals. 53
3.20 Comment rating distribution for comments from troll users and non-troll users in (a) YouTube and, (b) Slashdot. 54
3.21 Precision-recall curves for troll detection in the YouTube and Slashdot datasets. 55

4.1 Category distribution of (a) popular, and (b) tail queries. 64
4.2 Number of results (reported by YouTube) for (a) popular, and (b) tail queries. 65
4.3 Avg. no. of (a) views, (b) likes, (c) dislikes and (d) comments vs. video rank in the query results (for the popular and tail queries). 66
4.4 Average NDCG@10 for top-10 videos per feature for (a) popular, and (b) tail queries. 72
4.5 Fraction of queries for which a given feature yields the ranking with the highest NDCG@10 for (a) popular, and (b) tail queries. 72
4.6 $NDCG$ scores for the LETOR algorithms w.r.t. the number of features for the popular queries. 79
4.7 $NDCG$ scores for the LETOR algorithms w.r.t. the number of features for the tail queries. 80

5.1 Distribution of queries over the sentiment classes for different templates. 89
5.2 Sentiment distribution of (a) query result titles, and (b) query result snippets for the queries from each sentiment class. 92
5.3 Sentiment distribution of the clicked results for (a) positive queries, and (b) negative queries. We also show the fraction of the pages that are not found, i.e., not accessible online anymore. 93
5.4 (a) Mean sentivalue scores (from SentiWordNet) in each query class, (b) Distribution of average sentivalue scores of queries (from SentiWordNet) obtained from each template. 94
5.5 Distribution of query snippets' (a) objectivity, (b) positivity, and (c) negativity sentivalue scores (from SentiWordNet) in each query sentiment class. 95
5.6 Distribution of sentiment class annotations for each topic using queries submitted in (a) English, (b) German, and (c) Spanish. 98
5.7 Precision-recall curves and BEPs for (a) subjective vs. all, (b) positive vs. all, and (c) negative vs. all classifiers. 101
5.8 Query recommendation performance based on (a) in-house annotations, and (b) AMT annotations. 104
5.9 A toy example illustrating controversial topic detection: the procedure will output only "zen" as being controversial, as it yields very high variance in query sentiment scores and filter "zendaya", as its queries have less variance. 107

List of Tables

3.1 Descriptive statistics for the YouTube and Yahoo! News corpora. 23
3.2 Top-50 terms according to their MI values for accepted comments (with high comment ratings) vs. not accepted comments (with low comment ratings). 26
3.3 Examples of comments belonging to the categories *"accepted"* and *"unaccepted"*. 27
3.4 Top and Bottom-25 tags according to the variance of comment ratings for the corresponding videos. 34
3.5 Most probable terms for the top-5 and bottom-5 latent topics according to the comment rating variance of the corresponding videos. 35
3.6 Comment rating classification: BEPs for different training set sizes T and different rating thresholds. 38
3.7 Examples of comments belonging to the category *"seed comments"* (comments that received replies). 40
3.8 Basic statistics for seed and reply comments in the YouTube and Yahoo! News corpora. 41
3.9 BEPs for classification of seed comments vs. comments without replies. 44
3.10 Examples of comments belonging to the categories *"controversial"* and *"non-controversial"*. 46
3.11 Top-20 terms according to their MI values for controversial vs. non-controversial comments. 49
3.12 BEPs for controversial comment prediction. 50
3.13 Top-20 terms according to their MI values for troll vs. non-troll comments. 54

4.1 Query volume characteristics for the popular and tail queries. 61
4.2 Example popular and tail queries with the global monthly average search volume. 62
4.3 Metadata fields stored for each video. 62
4.4 Metadata statistics for the videos retrieved for the popular and tail queries. Lengths of the titles, tags and descriptions are in terms of the words (after stopword removal). 63
4.5 The list of all the basic and social features (F) employed in this work. . 70
4.6 Distribution of relevance labels. 71
4.7 Kendall's Tau values for the feature pairs computed over the top-10 rankings for the popular (upper diagonal) and tail (lower diagonal, shaded) queries (The value 0 means completely different rankings and 1 means equal rankings). Note that, the features f_{TCR} and f_{TPos} are available only for the tail queries. 74
4.8 Average $NDCG$@10 scores for LETOR algorithms using the basic and best-k features obtained with the GAS and MMR strategies for the popular and tail query sets (for bold cases, differences from the baseline are statistically significant). For GAS and MMR, we also denote the number of selected features (k) in parentheses. 78

5.1 List of controversial topics (along with the number of manually annotated queries per topic). 86
5.2 Templates for gathering queries (along with the number of manually annotated queries per template): queries for templates 1-5 are obtained using the query suggestion service, and those for template 6 are extracted from the AOL log. 88
5.3 Queries and sentiment categories for the topic "George Bush". 89
5.4 Top-20 (stemmed) query terms w.r.t. MI values for objective vs. subjective category (left) and positive vs. negative category (right). 90
5.5 Topics and the number of manually annotated queries (obtained via template 5) in each of the three languages (English, German and Spanish). 96
5.6 Examples of objective, positive and negative queries in each of the three languages (English, German and Spanish) for the topic "Iphone". . . . 97
5.7 Classification accuracy and AUC for the subjective vs. all classifiers trained with four different representations of the queries (*QAll* stands for *QTextTitleSnippet*). 100

5.8 Classification accuracy and AUC for the positive vs. all classifiers trained with four different representations of the queries (*QAll* stands for *QText-TitleSnippet*). 100
5.9 Classification accuracy and AUC for the negative vs. all classifiers trained with four different representations of the queries (*QAll* stands for *TextTitleSnippet*). 100
5.10 Search engine's suggestions (provided as "related queries" and "auto-completions", the latter are shown in italics) vs. opinionated suggestions for the query "economy is really bad". 105
5.11 Topics ranked with respect to the variance in sentiment scores of their queries. 108

1 Introduction

1.1 Motivation

The rapidly increasing popularity and data volume of modern Web 2.0 content sharing applications is based on their ease of operation even for unexperienced users, suitable tools for supporting collaboration and the attractiveness of shared content, e.g. images in Flickr, videos in YouTube, etc. These applications offer several social mechanisms for community interaction.

One of the most widespreaded mechanisms for community interaction in Web 2.0 sites is the possibility to comment on posted content and, in addition, to rate comments written by other users (see Figure 1.1). Comment ratings serve the purpose of helping the community in filtering relevant opinions more efficiently. Furthermore, because negative votes are also available, comments with offensive or inappropriate content can be easily skipped. Therefore, comments and associated ratings constitute a potentially interesting data source to mine for obtaining implicit knowledge about shared content as well as community interests and interaction behavior. The analysis of how users react to shared content and how they interact with each other is as important as the analysis of the content itself. Specifically, understanding these community dynamics has potential application to improving search and recommendation for content and comments based on these interactions.

In addition, such platforms allow their users to express themselves by rating the viewed objects (via clicking on the popular like/dislike buttons) and interacting with the other community members (also via the comments feature). This results in a vast amount of social signals associated with the shared content that may be exploited, among others, to improve the retrieval effectiveness. For instance, the user ratings for an object can serve as a global indicator of its quality or popularity (analogously to how the web graph features, such as PageRank, serve for the same purpose for the web pages), and the comments and other collaboratively formed data can facilitate and

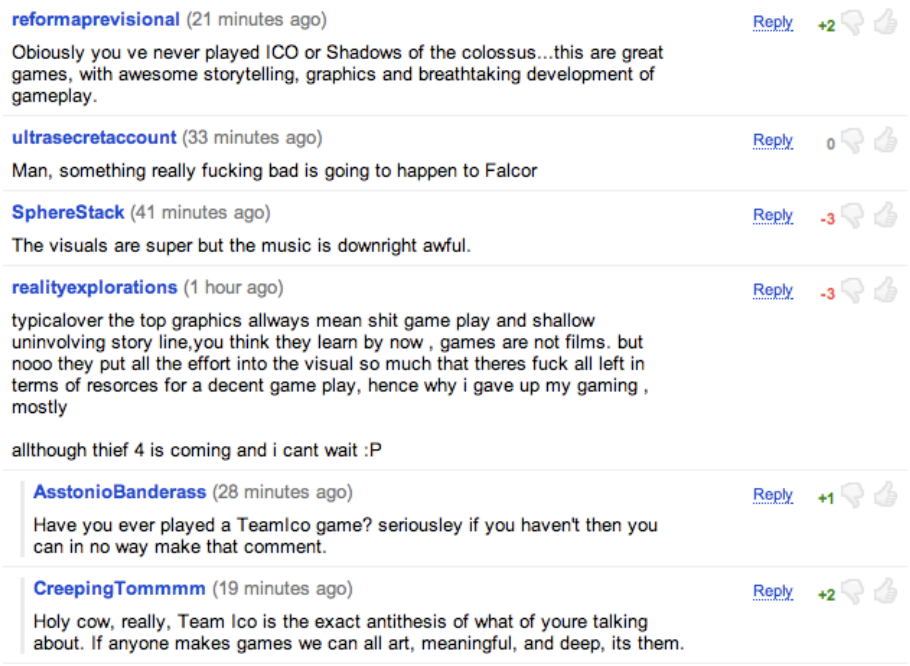


Figure 1.1: Examples of comments and ratings in YouTube.

enhance matching the shared content with the user queries. Despite the rapidly growing interest for Web 2.0 applications from both the industry and research communities, the impact of employing such mixed social signals within a large-scale search scenario is not fully explored.

As a motivating example for the potential gains of employing the social features during the retrieval process, consider the popular query "michelle phan" (a famous make-up instructor and product demonstrator). We obtained top-100 videos retrieved for this query from YouTube and re-ranked them, first, using the query-title similarity scores (based on the typical TF-IDF weighting model and Cosine measure). The upper row in Figure 1.2 shows the top-3 videos obtained at the end of this process and all having the title "michelle phan". The first video does not include Michelle Phan herself, but someone else who is imitating her, and hence, it is irrelevant. Indeed, the video is rated with a large number of dislikes. The third video is not highly relevant as well, as it contains a collection of her images compiled by someone else. In contrast, when we use a social feature based on the comment ratings, all the top-3 results are relevant to the query (see lower row in Figure 1.2), as these are the videos that are uploaded by her and/or including her. Also note that, the videos in the lower row are viewed an order of magnitude more times than those in the upper row. This anecdotal example demonstrates the promises and importance of social features especially for the multimedia-heavy platforms, such as YouTube and Flickr, that lack typical textual and link-based clues employed in the web search scenario. Specifically, given that the

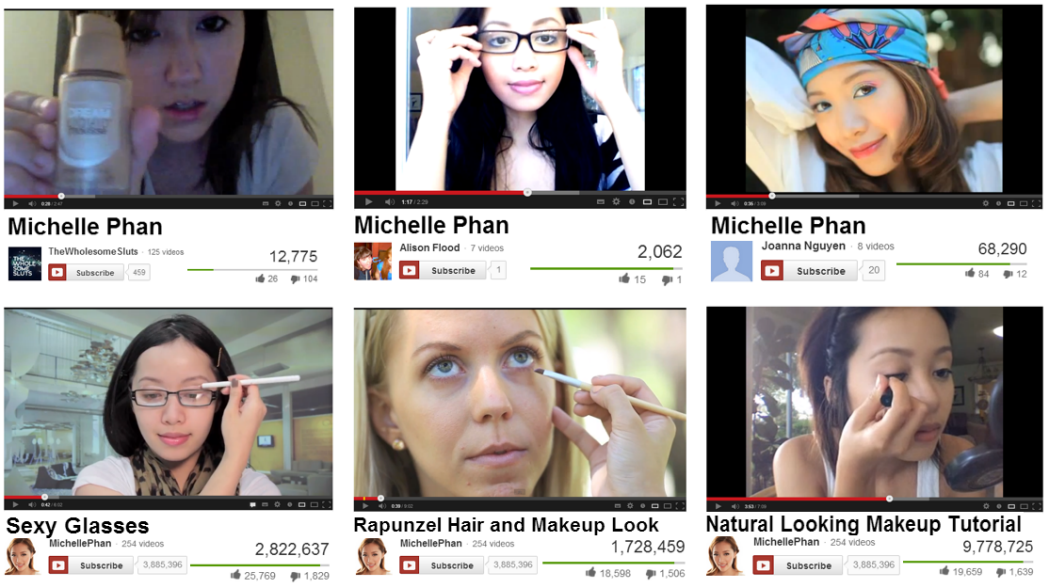


Figure 1.2: Top-3 videos for the query "michelle phan" re-ranked using the query-title similarity (upper row) and comment ratings (lower row).

content shared in such Web 2.0 platforms have shorter textual metadata (in comparison to the text in the web pages), such social features can prove to be useful to improve the retrieval quality.

Another type of feedback exists in the opinionated text appearing on the Web, with people discussing ideas and political issues, criticizing movies, reviewing books, or elaborating on features of their newly-bought camera. Not surprisingly, this content is not only appreciated by ordinary end users but also by professionals ranging from marketing and advertisement specialists to political strategists. The growing interest and demand for automatic analysis techniques and tools for opinionated digital texts have also fuelled research and led to various approaches in opinion mining and sentiment analysis [103]. While there exists a considerable body of literature on mining opinions from product reviews, blogs, Web search results, news articles and microblogs [103, 102, 129], another rich source of information, namely, Web search queries, has largely been overlooked. We anticipate that a non-trivial amount of Web queries that explicitly reflect opinions is issued to search engines, especially on controversial/popular topics in the society. For instance, when searching for the topic "abortion" using a major search engine, we are suggested not only a number of neutral queries such as "abortion facts" or "abortion statistics" but also queries that are in support of or against abortion (e.g., "abortion is right" vs. "abortion is morally wrong"). As these suggestions are usually based on real (and frequent) queries by other users, this provides clear evidence

that opinionated queries are not exceptional on the Web[1]. However, to the best of our knowledge, no previous work has attempted to characterize opinionated queries or detect and exploit the community sentiment in such queries.

1.2 Overview of Contributions

Comment-Centric Feedback The study of community interaction through the commenting feature provides the basis for many technological advances targeted at improving online communities by devising ways to encourage participation, facilitating access to relevant content and comments, and creating a safer and more appealing environment.

In this thesis, we will provide an in-depth analysis of comment-centric feedback found in online community websites. Although there is a large body of work that investigates and leverages comments in various social platforms (e.g. [108, 115, 56, 131]), to the best of our knowledge, we are the first to consider comment ratings as first class citizens and exploit them to gain a better understanding of comments, content, and users.

We consider the analysis of two different datasets obtained from popular online communities, namely YouTube[2], an online community centered around user-generated content, and Yahoo! News[3], which is centered around editorial and curated content. YouTube is the most popular video sharing site, and traffic to/from this site accounts for over 20% of the web total and 10% of the whole internet [35], and comprises 60% of the videos watched on-line [62]. In 2014 YouTube reported having over 1 billion unique visitors each month and over 100 hours of videos beeing uploaded each hour [147]

Yahoo! News is one of the leading news aggregators, and attracts a large number of users who follow news stories around the world and comment on them with over 138 million distinct yearly visitors. Both platforms provide a rich sample of comments and associated metadata for a variety of content objects from a large pool of users. In addition to providing insights into several properties of comments and ratings, we also identify differences between the two datasets that occur due to behavioural differences of their corresponding communities. The work in this section follows three main types of analyses, each of them exploiting comment-centric feedback existing in online communities.

First, our analysis considers the comments themselves. We analyze how the used language and polarity of opinions in comments influence their perceived value by the

[1] Note that an opinionated query may not necessarily express the personal view of the user who submitted it.

[2] http://www.youtube.com

[3] http://news.yahoo.com/

rest of the community, triggers further discussion, or divides the community. In addition to shedding light into fundamental relationships of comments and their ratings, machine learning models trained using comment-centric information can directly enhance comment browsing by promoting comments that are likely to receive high ratings in the future, initiate a discussion thread, or create controversy within the community. This can help to improve user satisfaction by enabling smart comment ranking methods and ultimately fuel user interaction and engagement with the underlying system. Intelligent feedback mechanisms can also benefit from this knowledge to provide users with guidelines for commenting behaviour well received by and useful to other users.

Second, we analyze how the particular features of shared content influences community interaction and leads to polarized discussions. Detecting content that polarizes the community can be useful for providing an additional facet for retrieval and exploratory search, as is very commonly sought by a large number of users. This can be also applied for understanding community dynamics and studying the evolution of controversy along time.

Finally, we analyze individual users in the community that negatively influence the normal dynamics, compromising the experience of fellow users. We study textual content of comments and their ratings for detecting *trolls*, i.e., "users who post disruptive, false or offensive comments in online communities to fool and provoke others [79]". Because of their disruptive nature, the presence of trolls can highly compromise user engagement in online communities. We study the ability of machine learning tools to detect trolls to allow for early detection and excision from the system.

Social Feedback Web search engines, taking their fair share from the Web 2.0 wave, have taken steps towards a more "social" search. Bing announced its "Liked Results" feature, which, in a nutshell, annotates the result URLs with the names of the searchers' friends who *liked* these URLs publicly or shared them via Facebook. This evolved latter to Bing's social search feature that is provided via a sidebar on the search results page. This sidebar subsumes a wide range of social functionalities, most strikingly identifying your Facebook friends, who might know about your query, based on their *likes*, profile information, shared photos, etc. During this time, Google also released its "Search plus Your World" feature that enriches algorithmic results with pages, photos and posts from the searchers' Google+ social network. While all of these recent developments imply the importance of social signals in search, the details, i.e., how exactly and to what extent such signals can be exploited in ranking query results, are not disclosed due to the highly competitive nature of the market.

The research community showed keen interest in analyzing and exploiting the rich content shared in Web 2.0 platforms. Some earlier studies attempted to investigate the retrieval potential of the social signals, specifically comments, in isolation (e.g., [146, 96, 108]). However, to the best of our knowledge, there is no study that systematically and exhaustively investigates the impact of a rich set of social signals on the retrieval performance in a realistic and state-of-the-art framework.

How useful are social signals to improve the retrieval effectiveness? In this part, we seek an answer to this central question. While doing so, we focus on the keyword-based video search for YouTube video sharing site. *Social features*, as we call in this thesis, refer to the information that is created by some explicit or implicit user interaction with the system (such as views, likes, dislikes, favorites, comments, etc). In contrast, we call the features that would be typically involved in a keyword search scenario, such as the textual similarity of the user queries to the video title, tags and description (i.e., metadata fields provided by the content uploader) as the *basic features*. Our work essentially explores whether the social features in combination with the basic features can retrieve more relevant videos; and if this is the case, which social features serve the best. While our choice of YouTube is based on the availability of a rich set of social features in this platform, we believe that the findings are applicable to the text, image and/or video search in other platforms that support similar kinds of features.

Community Sentiment in Web Queries To the best of our knowledge, we are the first to provide a detailed analysis of sentiment in Web queries on controversial topics. To this end, we employ a number of different query templates on the query suggestion service of a major search engine as well as a publicly available query log to obtain a large and representative sample of real user queries. Using this dataset, we conduct manual and lexicon-based analysis of sentiments in the queries, and provide answers to various research questions: To what extent can Web queries include opinions (this may or may not reflect the query issuers own opinion)? To what extent is sentiment in the queries mirrored in retrieved results and user clicks? Is sentiment in the queries correlated with the geographical locations of users?

Secondly, we study the applicability of state-of-the-art sentiment analysis methods (including both lexicon-based and machine learning based methods) for detecting the sentiment of the queries. Query texts exhibit inherently different characteristics in comparison to classical corpora used for sentiment analysis (i.e., news stories, blogs, product reviews, comments, and even tweets). In this work, we use features obtained from the top-ranked result titles and snippets, as well as the pure query text, while applying and evaluating the current sentiment detection techniques for this new source

of data with its unique characteristics. The performance is evaluated on more than 7,651 human annotated queries for 50 controversial topics.

As a final contribution of this chapter, we employ our query sentiment detectors in two use case scenarios, namely, *query recommendation* and *controversial topic discovery* (for trend analysis). In extensive user studies including both in-house participants and workers from a crowdsourcing platform we show the viability of sentiment detection for both applications.

1.3 Outline of the Thesis

The rest of this thesis is organized as follows. **Chapter 2** introduces a couple of state-of-the-art techniques which will be used for leveraging different types of community feedback.

In **Chapter 3** we provide an in-depth analysis of comment-centric feedback found in two online media websites, YouTube and Yahoo! News. We explore the applicability of machine learning and data mining techniques to detect the acceptance of comments, comments likely to start discussion threads, controversial and polarizing content, and users showing offensive commenting behavior. The work reported in Chapter 3 is contained in the following publications:

- Siersdorfer, S., Chelaru, S., Nejdl, W., and San Pedro, J. How useful are your comments?: analyzing and predicting youtube comments and comment ratings. In *Proceedings of the 19th International Conference on World Wide Web* (2010), WWW '10, ACM, pp. 891–900
- Siersdorfer, S., Chelaru, S., San Pedro, J., Altingovde, I. S., and Nejdl, W. Analyzing and mining comments and comment ratings on the social web. *ACM Transactions on the Web 8*, 3 (July 2014), 17:1–17:39

In **Chapter 4** we study the social feedback that is associated with the top-ranked videos retrieved from YouTube for real user queries. We investigate the effectiveness of individual social features for video retrieval and the correlation between the features. Finally, we investigate the impact of the social features on the video retrieval effectiveness using state-of-the-art learning to rank approaches and two feature selection strategies. The work presented in Chapter 4 was published in:

- Chelaru, S., Orellana-Rodriguez, C., and Altingovde, I. S. Can social features help learning to rank youtube videos? In *Proceedings of the 13th International Conference on Web Information Systems Engineering* (2012), WISE '12, Springer-Verlag, pp. 552–566

- Chelaru, S., Orellana-Rodriguez, C., and Altingovde, I. How useful is social feedback for learning to rank youtube videos? *World Wide Web Journal* (2013), 1–29

In **Chapter 5** we present and in-depth analysis of user and community sentiment in Web queries. Furthermore, we build various query sentiment classifiers and analyze their performances. Finally, we present the virtue of query sentiment detection in two different use cases. The results reported in Chapter 5 have appeared in:

- Chelaru, S., Altingovde, I. S., and Siersdorfer, S. Analyzing the polarity of opinionated queries. In *Proceedings of the 34th European Conference on IR Research* (2012), ECIR '12, Springer-Verlag, pp. 463–467

- Chelaru, S., Altingovde, I. S., Siersdorfer, S., and Nejdl, W. Analyzing, detecting, and exploiting sentiment in web queries. *ACM Transactions on the Web 8*, 1 (Dec. 2013), 6:1–6:28

We conclude our work in **Chapter 6**, where we summarize our main contributions and discuss the main future research directions.
Additional related results and ideas have been reported in a couple of other works:

- Chelaru, S., Herder, E., Djafari Naini, K., and Siehndel, P. Recognizing skill networks and their specific communication and connection practices. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media* (2014), HT '14, pp. 13–23

- Rokicki, M., Chelaru, S., Zerr, S., and Siersdorfer, S. Competitive game designs for improving the cost effectiveness of crowdsourcing. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14 (Accepted Paper)

- Demartini, G., Siersdorfer, S., Chelaru, S., and Nejdl, W. Analyzing political trends in the blogosphere. In *Proceedings of the Fifth International Conference on Weblogs and Social Media* (2011), ICWSM '11

- Demartini, G., Siersdorfer, S., Chelaru, S., and Nejdl, W. Exploiting the blogosphere to estimate public opinion in the political domain. In *GLocal Report* (2011)

- Chelaru, S., Stewart, A., and Siersdorfer, S. Exploiting an inferred comment graph for clustering videos in youtube. In *GLocal Report* (2011)

2 Background

In this chapter we provide an overview of state-of-the-art techniques which will be used for leveraging different types of community feedback in a wide range of problem settings. First, we describe some key concepts and approaches for classifying text documents. In this thesis we will apply these techniques in various novel contexts to automatically classify (1) comments according to their community feedback and (2) queries according to their opinions. Then, we discuss the notion of sentiment analysis, as well as sentiment classification. Our work exploits these techniques in various studies regarding sentiment and opinions across comments and query logs. In addition, we describe the notion of social features in the context of social content. Finally, we provide an overview of learning to rank methods, which will be used to evaluate the effectiveness of social features for the video retrieval effectiveness.

2.1 Text Classification

Text classification refers to the supervised machine learning approach in which a model capable to distinguish documents belonging to different *classes* (also known as *categories*) is learned. In order to learn the classification model, a set of labeled *training* documents is required. The learned model can be applied to predict the class labels for a set of *test* documents, for which the class is unknown. Training and test documents, which are given to the classifier, are represented as multidimensional vectors $\vec{d} = (d_1, ..., d_m)$. These vectors can, for instance, be constructed using TF or $TF \cdot IDF$ weights which represent the importance of a term for a document in a specific corpus [10, 120]. In this thesis, we used different state-of-the-art text classification approaches.

Support vector machines (SVMs) were introduced by Vapnik et al. [134, 38] and are considered to be highly accurate methods for a various set of classification tasks [48, 104, 49, 80]. Manning et al. [92] provide a formalization of the SVM classification

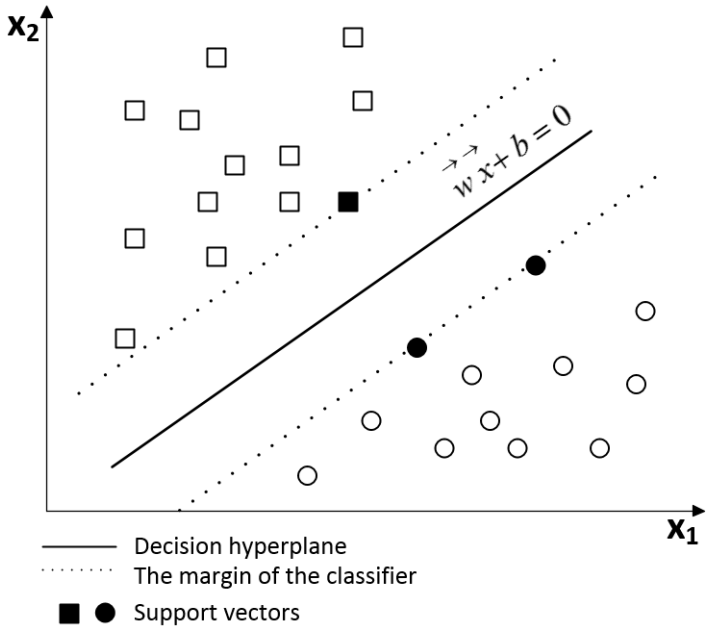


Figure 2.1: Maximum margin decision hyperplane for a linear SVM.

method in the context of text classification. Given a set of n training documents $D = \{(\vec{d_i}, y_i)\}$ with $\vec{d_i} \in R^m$ and $y_i \in \{-1, 1\}$ the corresponding class labels, we make the assumption that the training data is linearly separable. The linear SVM method aims at finding a hyperplane $\vec{w} \cdot \vec{x} + b = 0$ that separates the set of positive training documents from the set of negative documents with a *maximum margin*. Because of its final role on deciding to which class a new document should be assigned to, the separating hyperplane is also known as the *decision hyperplane* [92] or *decision surface* [1].

The hyperplane is defined by the *intercept term* b and a normal vector $\vec{w}$ (called *weight vector*) that is perpendicular to it. In order to find an optimal hyperplane which separates the positive training instances from the negative ones, we are required to solve *the SVM minimization problem* [92]:

- Find $\vec{w}$ in order to minimize $\frac{1}{2}\vec{w} \cdot \vec{w}$
- subject to $y_i(\vec{w} \cdot \vec{d_i} + b) \geq 1$, $\forall i \in \{1, ..., n\}$, where $y_i \in \{-1, 1\}$

For a new previously unseen test document $\vec{d}$, the SVM predicts its class by checking whether it lies on the "positive" or the "negative" side of the separating hyperplane:

$$class(\vec{d}) = \begin{cases} 1, & \text{if } (\vec{w} \cdot \vec{d} + b) > 0 \\ -1, & \text{otherwise.} \end{cases}$$

Figure 2.1 shows the maximum margin decision hyperplane for a dataset containing a set of positive (squares) and negative (circles) instances in a 2-dimensional feature space.

Traditionally, SVMs have been used for *binary classification* scenarios, but they can be adapted for a multiclass case. In this thesis, we build binary and multiclass classifiers using the SVM formulation implemented in the LIBSVM package [26] as well as the SMO variant and the L2-loss linear methods implemented in the well-known Weka library [68].

The **Naive Bayes** classification method (e.g., [93, 92]) is a probabilistic approach which makes the "naive" assumption that the term attributes are conditional independent of each other, given the document classes. For a test document d, the Naive Bayes approach can estimate the probability of d to belong to class c_j, where $j \in \{1, ..., k\}$ based on: (1) the prior computed conditional probability of each term w_k in d to occur in a document of class c_j and (2) the fraction of training documents belonging to class c_j.

The **Logistic Regression** [14] is a statistical and probabilistic classification method which learns a logistic (sigmoid) function in order to predict binary outcomes. In this thesis, we build classifiers using the Multinomial Naive Bayes and the Simple Logistic Regression methods from the Weka library.

2.2 Sentiment Analysis

Sentiment analysis is the area which tackles the problem of understanding opinions and their polarity (e.g. "positive" vs. "negative" vs. "neutral") in textual content. Much of the work in this field has focused on the task of **sentiment classification** [104, 129]) which deals with the problem of *automatically* assigning opinion values to documents or topics using various text-oriented and linguistic features. Figure 2.2 presents two examples of positive and negative opinionated text documents (Amazon Reviews) where users express their opinion towards the movie *Madagascar 3:Europe's most wanted*. The problem of classifying movie reviews, based on the sentiment expressed into positive or negative using various classification methods was first studied by Pang et al. [104]. Their results showed that the SVM approach using various textual features provides the highest accuracies.

Recent work in the area makes use of annotated lexical resources such as Senti-

Figure 2.2: Examples of positive (top) and negative (bottom) opinionated movie reviews (from Amazon).

WordNet [53] or SentiStrength [127] to improve classification performance (e.g., the former thesaurus is employed in [47]). Cross-domain sentiment classification was studied, for instance, in [102] where spectral graph analysis is used to infer links between domain-independent and domain-specific terms. In the position paper [100] the authors provide an overview of challenges in opinion retrieval that arise due to the highly context-dependent character of opinions expressed in Web pages. The authors propose a grammar-based approach to account for opinion-related contexts on a sentence level. There are several works that make use of sentiment thesauri for exploratory studies. For instance, in [120] we use SentiWordNet to analyze sentiment in YouTube comments and the relationship between sentiment and comment ratings. In [85] the SentiStrength resource is leveraged for studying sentiment in Yahoo! Answers with respect to temporal and demographic aspects. Vural et al. [138] employs a sentiment thesaurus to guide a focused crawler for discovering opinionated web content.

SentiWordNet [53] is an enhanced lexical resource which was built on top of WordNet [55] and can be exploited in various sentiment analysis tasks. WordNet is a thesaurus containing descriptions of terms and semantic relationships between terms (examples are hypernyms: "car" is a subconcept of "vehicle" or synonyms:"car" describes the same concept as "automobile"). WordNet distinguishes between different part-of-speech types (verb, noun, adjective, etc.) A *synset* in WordNet comprises all terms referring to the same concept (e.g. $\{car, automobile\}$). In SentiWordNet a triple of three *senti values* (pos, neg, obj) (corresponding to positive, negative, or rather neu-

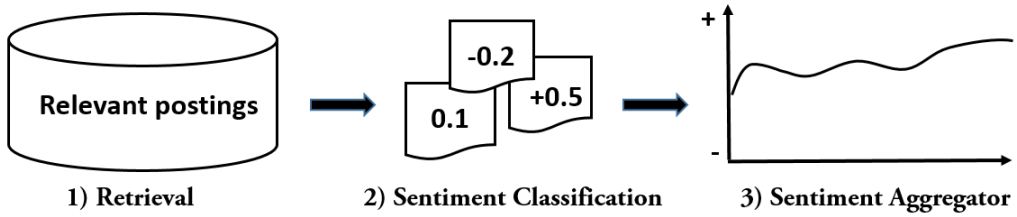


Figure 2.3: The Opinion Mining and Aggregation System developed in [45].

tral sentiment flavor of a word respectively) are assigned to each WordNet synset (and, thus, to each term in the synset). The sentivalues are in the range of $[0, 1]$ and sum up to 1 for each triple. For instance $(pos, neg, obj) = (0.875, 0.125, 0)$ for the term "awesome" or $(0.125, 0.75, 0.125)$ for the term "wrong". Sentivalues were partly created by human assessors and partly automatically assigned using an ensemble of different classifiers (see [52] for an evaluation of these methods).

In [45] we developed a framework able to capture *the temporal development of opinions* on politicians. In order to estimate the public opinion towards an entity, the system follows three steps. In the first step, entity-relevant blog postings are retrieved. Next, we make use of linear SVM classifiers and the SentiWordNet resource to assign a sentiment value to each relevant post. In the last step, the system employs different aggregation methods over the sentiment scores to estimate the development of opinions over a time frame. Figure 2.3 shows the mining, sentiment classification and aggregation steps performed by the system.

2.3 Social Content and Features

Web 2.0 platforms and social networks received a widespread attention in the last decade. Musial and Kazienko provide an in-depth survey of the social networks from a broad perspective including the sites directly intended for such networking purposes (such as MySpace and LinkedIn) and other platforms where the users form an implicit community via interacting with the system (such as Flickr, YouTube, etc.) [99]. Cheng et al. provides a large-scale analysis of the content in YouTube and provide statistics related to the videos, such as the distribution of categories, duration, size, bit rate and popularity [36]. An analysis of the video characteristics, such as the popularity distribution and evolution over time are addressed in [25]. Vavliakis et al. [135] compare YouTube and two other data sharing platforms in terms of several factors and identify the correlations between these factors via regression analysis. Various properties of the comments posted for YouTube videos are analyzed in [128]. In an out-of-the-

laboratory study aiming to shed light on how people find and access videos on the Web, Cunningham et al. [39] also discuss under what circumstances the participants benefit from the comments.

Social features, as we call in this thesis, refer to the information that is created by some explicit or implicit user interaction with the system. In this sense, we derive the social features from *raw features* (also named *social interactions* in [69]) such as views, likes, dislikes, favorites, comments, or other similar available data. In contrast, we call the features that would be typically involved in a keyword search scenario, such as the textual similarity of the user queries to the video title, tags and description (i.e., metadata fields provided by the content uploader) as the *basic features*. Among the social features associated with the shared content, the lion's share of research interest is devoted to the user comments due to their potential to improve the performance in several scenarios. In a recent survey, Potthast et al. [108] categorize the comment related tasks as comment-targeting and comment-exploiting. The works that aim to rank [75] or diversify the comments [61] and predict their ratings [120] fall into the former group. In the latter category, there is a large body of works that utilize the comments for various purposes, such as summarizing the blog posts [76], classification of YouTube videos [56], predicting the content popularity [96, 131, 145, 74] and recommending the related content items [118].

2.4 Learning to Rank (LETOR)

In the last years, traditional ranking approaches based on the manually designed ranking functions (such as BM25, TF-IDF, etc.) are replaced or complemented by the rankers built by machine learning strategies [27]. The commercial web search engines typically apply a two-stage ranking process where a candidate set of documents is identified using a traditional yet relatively inexpensive approach in the first stage [20]. Next, these candidate documents are re-ranked using a **learning-to-rank (LETOR)** strategy based on several hundreds of features. In a typical LETOR framework, a machine learning algorithm is trained using a set of triples of (q, F, r), where q is the query id, F is the m-dimensional feature vector for a result object retrieved for q, and r is the relevance score. The learned model is used to predict the relevance score for each pair (q, F) in the test set, which is then sorted with respect to these predicted scores. The success of the ranking model is evaluated using measures like the *Mean Average Precision*(MAP) [10], *Normalized Discounted Cumulative Gain*(NDCG) [27] or the *Expected Reciprocal Rank* (ERR) [28]. Figure 2.4 shows an example of a LETOR

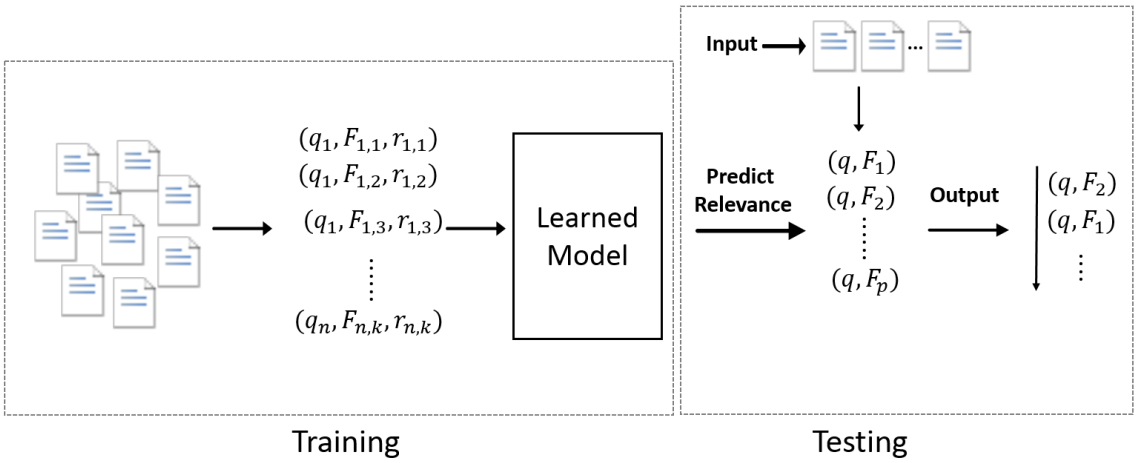


Figure 2.4: A Learning to Rank framework.

framework, where the learned model is used to predict the query-document relevance scores for a set of p input documents.

A variety of LETOR approaches appear in the literature, for which we refer to [88] as an exhaustive survey. These approaches are broadly categorized into three categories, namely, *point-wise*, *pair-wise* and *list-wise* depending on their loss function. In this thesis, we employ state-of-the-art representatives from each category:

- *RankSVM*: This pair-wise approach extends traditional SVM by utilizing instance pairs and their labels during training. One popular implementation is provided by Joachims [81].

- *RankBoost*: First introduced by [58], this algorithm also employs a pair-wise technique. It uses a well known machine learning technique called *boosting* [116] in order to combine partial weak rankings obtained based on different ranking features.

- *ListNet*: Instead of taking documents pairs as the instances, list-wise approaches exploit the lists of documents during the learning. In particular, ListNet [22] is based on the Neural Networks and employs the Gradient Descent algorithm [19] in the optimization stage.

- *CoordinateAscent*: This is again a list-wise linear model which uses coordinate ascent technique that optimizes multivariate objective functions by sequentially doing optimization in one dimension at a time [95]. For RankBoost, ListNet and Coordinate Ascent methods, the RankLib package[1] provides a robust implemen-

[1] http://people.cs.umass.edu/~vdang/ranklib.html

tation and therefore was used in our work.

- *Gradient Boosted Regression Trees (GBRT)*: This is a simple yet very effective point-wise method for learning non-linear functions [59], claimed to be the current state-of-the-art learning paradigm [98].

- *Random Forests (RF)*: Random Forests is a point-wise ranking approach based on the bagging technique, i.e., applying the learning algorithm multiple times on different subsets of the training data and averaging the results [98]. RF is proposed as a low-cost alternative to GBRT with the additional advantage of being very resistant to over-fitting.

- *Initialized Gradient Boosted Regression Trees (iGBRT)*: This approach uses the predictions from the RF algorithm as a starting point for the *GBRT* algorithm [98]. The RT-Rank library[2] provides reliable implementations for the *GBRT*, *RF* and *iGBRT* methods.

In addition to the learning algorithms, **feature engineering** is an important aspect within a LETOR framework. **Feature selection** is a well-known approach in machine learning for enhancing the accuracy (e.g., by preventing over-fitting) of the learned model and efficiency of the learning process [60, 42, 77]. Geng et al. [60] address the vitality of the feature selection issue for machine learning based approaches to the ranking problem and propose a greedy feature selection strategy. Formally, given a set of features $\{f_1, ..., f_m\}$ and the target number of features, $1 \leq k \leq m$, the goal is selecting the k features that would yield the maximum performance for a LETOR algorithm. Each feature is associated with an importance score, $Imp(f)$, which is an indicator of the retrieval effectiveness of f. Furthermore, for each feature pair (f_i, f_j), similarity of their top-N rankings, is computed. The optimization problem is defined as choosing a set of k features that maximizes the sum of the feature importance scores and minimizes the sum of the similarity scores between any two features.

In the last few years, large search companies such as Microsoft, Yahoo!, and Yandex released benchmark datasets for so-called **LETOR challenges**. However, the features employed in these datasets are only broadly described (e.g., [27, 110]) and the actual feature names in the data are never disclosed, making it impossible to analyze the importance/utility of a particular feature or class of features. To overcome this latter difficulty, a recent study presents a new dataset based on the data collected from a commercial Chilean search engine, TodoCL [4]. The dataset includes 79 queries

[2] http://research.engineering.wustl.edu/~amohan/

with 3,119 relevance assessments and a total of 29 features. In [90], Macdonald et al. employ the official queries and their top-ranked documents from the TREC collections to analyze the usefulness of the query features in a LETOR setup.

Other studies addressed the image and/or video retrieval within a LETOR framework. For instance, Jain and Varma exploit user click data in addition to the textual and visual features for query-dependent image re-ranking [78]. In [94], a new LETOR approach is proposed to rank large-scale image or video collections, and evaluated using various visual features obtained from the TRECVID 2007 collection. Davidson et al. discuss that various social signals are employed in the video recommendation system in use at YouTube, which serves as a further evidence for the potential of employing such features for the retrieval purposes [43].

Finally, a couple of works have addressed the problem of efficiency in the context of learning to rank. Wang et al. [139] propose a framework where models are learned using metrics able to capture the tradeoff between the expected efficiency and effectiveness of the learned functions. In [130] Tonellotto et al. analyse dynamic document pruning strategies such as WAND [17], a method which discards the documents unlikely to be retrieved in the top-k results. The authors show that by applying query performance predictors [73], the WAND method can increase the efficiency and maintain a high effectiveness for particular types of queries.

3 Comment-Centric Feedback on the Social Web

An analysis of the social video sharing platform YouTube and the news aggregator Yahoo! News reveals the presence of vast amounts of community feedback through comments for published videos and news stories, as well as through meta ratings for these comments. This chapter presents an in-depth study of commenting and comment rating behavior on a sample of more than 11 million user comments on YouTube and Yahoo! News. In this study, comment ratings are considered first class citizens. Their dependencies with textual content, thread structure of comments, and associated content are analyzed to obtain a comprehensive understanding of the community commenting behavior. Furthermore, this chapter explores the applicability of machine learning and data mining techniques to detect acceptance of comments by the community, comments likely to trigger controversy and discussions, controversial content, and users exhibiting offensive commenting behavior.

3.1 Related Work

A review of previous literature reveals a number of works that have been leveraging user comments in a wide range of different problem settings. One relevant application of user comment mining is the enhancement of retrieval mechanisms in online communities and social platforms [108]. Mishne and Glance [96] investigate the impact of comments on the retrieval performance for weblogs. They find that while involving comment text in scoring does not help to improve precision, it allows for retrieving both relevant and highly discussed blog posts as an alternative to retrieving only relevant answers. In [146], the authors demonstrate the potential of comments for improving the effectiveness in a known-item retrieval scenario for YouTube. In [115] user comments are leveraged to determine the visual quality of images and to compute an aesthetic-

aware re-ranking of image search results. Agichtein et al. [2] make use of lexical and social graph characteristics of comments and commenters in the network to find high quality content in the popular community answering system Yahoo! Answers.

Further tasks that make use of comment content include summarization of blog posts [76], prediction of video categories in YouTube [56], identification of political orientation in news articles based on comment sentiments [105], analysis and prediction of the popularity of the commented items (e.g., [96, 131, 145]), and recommendation of related content based on commented items [86, 118]. In none of these works, comment ratings are analyzed or taken into account in the first place.

There is a body of work on analyzing product reviews and postings in forums. In [41] the dependency of helpfulness of product reviews from Amazon users on the overall star rating of the product is examined and a possible explanation model is provided. "Helpfulness" in that context is defined by Amazon's notion of how many users rated a review and how many of them found it helpful. Lu et al. [89] use a latent topic approach to extract rated quality aspects (corresponding to concepts such as "price" or "shipping") from comments in eBay. In [143] the temporal development of product ratings and their helpfulness and dependencies on factors such as number of reviews or effort required (writing a review vs. just assigning a rating) are studied. The helpfulness of answers on the Yahoo! Answers site and the influence of variables such as required type of answer (e.g. factual, opinion, personal advice), topic domain of the question or "a priori effect" (i.e. Did the inquirer conduct some a priori research on the topic?) is manually analyzed in [70]. Kim et al. [83] rank product reviews according to their helpfulness using different textual features and meta data. However, they report their best results for a combination of information obtained from the star ratings (e.g. deviation from other ratings) provided by the authors of the reviews themselves; this information is not available for all sites, and in particular not for *comments* in YouTube and Yahoo! News. Weimer et al. [141] make use of a similar idea to automatically predict the quality of posts in the software on-line forum *Nabble.com*. In comparison, this thesis focuses on *community ratings for comments and discussions* rather than product ratings and reviews.

Only a few recent works focus directly on comment ratings. In [75] a regression model is proposed for ranking comments from the social news aggregator *Digg.com* based on the community ratings. The authors studied the impact of different comment features like visibility, reputation of the comment authors, and the actual content of the comments. In [40], the authors propose a multi-objective comment ranking strategy for 200 articles, each with 50 comments from Yahoo! News. While we also apply machine learning for comment rating prediction in YouTube (cf. Section 3.3.5), our analysis

and scenarios presented here cover a much wider scope than solely predicting ratings. In [97] unsupervised, semi-supervised and active learning strategies are employed on the user-comment graph to correct the bias in comment ratings. The analysis of bias in ratings is not in the scope of our work, though we consider their findings complementary to our research directions.

Detecting abusive users (e.g., spammers, vandals, or trolls) is another topic that has recently drawn a lot of attention in the context of social and collaborative platforms such as forums or wikis. For the specific case of comments, an earlier study [136] proposes to use associative classification to separate "good" comments from "bad" ones in order to enable automatic moderation in Slashdot, a popular technology news web site. The proposed classifier uses features based on the comments content and the social network (fans, friends, etc.) of the commenter. In another study addressing the same problem [108], the features used for classification represent the comment quality, and include comment length, readability, frequency of vulgar terms, etc. Our approach presented here differs from these works in that we detect troll *users* instead of individual comments. In [79], troll users in Slashdot are detected using global and node-level social graph characteristics. In contrast, we use a bag-of-comments model for users to classify the trolls. We further show that comment ratings can be a good indicator for detecting trolls.

Works on predicting discussion threads usually aim at detecting the content items (such as news articles [131, 126], tweets [114], or forum posts [113]) that are likely to attract comment replies. In [96], machine learning methods are used to identify disputative threads and in [37], a framework is developed for characterizing the interestingness of threads based on their themes and participants. In [63] the authors focus on the Slashdot network and repurpose a h-index as an effective metric of content controversy, where the number of nested replies for each comment is used as the h-index equivalent of *number of citations* for each paper. In contrast to these works, we predict the individual *comments* that are likely to attract other comments and start a discussion thread.

3.2 Data Gathering, Methods and Characteristics

3.2.1 Data Gathering

The research conducted in Sections 3.3, 3.4 and 3.5 was based on data gathered from two highly popular, community-oriented websites: YouTube and Yahoo! News. In Section 3.6 we will introduce an additional dataset gathered from Slashdot [1] which will

[1] http://www.slashdot.org/

be only used in the context of troll detection.
YouTube is a video sharing platform where users can upload their own videos and watch, rate and comment on other users' content. At Yahoo! News users can follow news stories around the world and comment on them. Both platforms provide tools for replying other users' comments and rating them via like/dislike buttons.

***YouTube* collection** We used the YouTube search engine to create this first collection by formulating textual queries. We selected our set of seed queries from Google's Zeitgeist archive from 2001 to 2007. We obtained a total of 756 different keywords. The top 50 results for each query were collected. We then extended this set using YouTube's "related videos" option over a sample of the already collected videos chosen uniformly at random. This scraping methodology aims at mimicking the typical user interaction with YouTube. For each selected video we gathered the first 500 comments (if available), along with contextual metadata, including authors, timestamps and comment ratings. YouTube computes comment ratings by counting the number of likes ("thumbs up") and dislikes ("thumbs down"), which correspond to positive and negative votes by other users. In addition, for each video we collected metadata such as title, tags, category, description, upload date as well as statistics provided by YouTube such as overall number of comments, views, and video rating. The complete collection had a final size of $67,290$ videos and over 6 million comments.

***Yahoo! News* collection** In order to form our second collection, we first collected all stories published in the Yahoo! News RSS feed between September and December 2011. For each story, we crawled all available comments along with their ratings (i.e., the number of likes and dislikes per comment) and replies, as well as associated meta data including the authors of comments, locations (if stated in the author profile), and timestamps. This process yielded a collection of 5.4 million comments for $27,000$ news stories.

Our rationale for using these two datasets is to cover different types of social media applications with different underlying incentives and commenting behavior. Information shared in these two online communities is not just different in terms of modality, but also in the particular characteristics that determine its relevance to users: videos in YouTube are commonly retrieved by specific queries issued by users, while news stories are mainly browsed in inverse chronological order in each of the predefined categories of the site. We are aware that the differences in the crawling methods used to collect each of the datasets produces two collections with intrinsically distinct characteristics. However, both crawling strategies aim at replicating the common retrieval interaction

Table 3.1: Descriptive statistics for the YouTube and Yahoo! News corpora.

	YouTube	Yahoo! News
Mean #comments	261.75	140.94
Median #comments	13	3
Max. #comments	128,307	48,051
Stddev. #comments	2,053.84	866.43
Mean #words	8.20	15.68
Median #words	5	9
Mean #sentences	1.82	2.76
Median #sentences	1	2
Mean rating	0.61	1.39
Median rating	0	0
Stddev. rating	8.42	10.95
Max. rating	4,170	4,327
Min. rating	-1,918	-1,018

of users in each website and allow for comparing the particularities of comments in two widely used online communities.

3.2.2 Data Characteristics

In Table 3.1, we provide descriptive statistics about our collections. For Yahoo! News, we observe a mean value of $\mu_{comm} = 140.94$ (median value of 3) comments per story, whereas for YouTube the mean number of comments per video is $\mu_{comm} = 261.75$ (median value of 13). These figures reflect the actual number of comments as reported by the corresponding systems, and not according to the number of crawled comments. The difference in the average number of comments is not unexpected: as news stories are updated rapidly, they are actively accessed only for a short time. In particular, top news stories are archived for 7 days only, enforcing a natural limitation on the number of comments a story can get. On the other hand, YouTube videos feature a longer life span, allowing further comments to be added by the viewers. To give an example, at the time of crawling the most commented news story had a total of $48,051$ comments, while the most commented YouTube video ("Miss Teen USA 2007 - South Carolina answers a question") in our collection attracted $128,307$ comments[2]. The Yahoo! News article with the highest number of comments[3] is reporting about incidents between the Muslim Community and the police in New York; these incidents were highly debated in the US. In contrast, comments posted for news stories are almost twice as long as those posted for videos, both in terms of the number of words and the number of sentences,

[2] http://www.youtube.com/watch?v=lj3iNxZ8Dww

[3] http://news.yahoo.com/blogs/new-york/muslims-police-scuffle-rye-playland-over-amusement-park-123309825.html

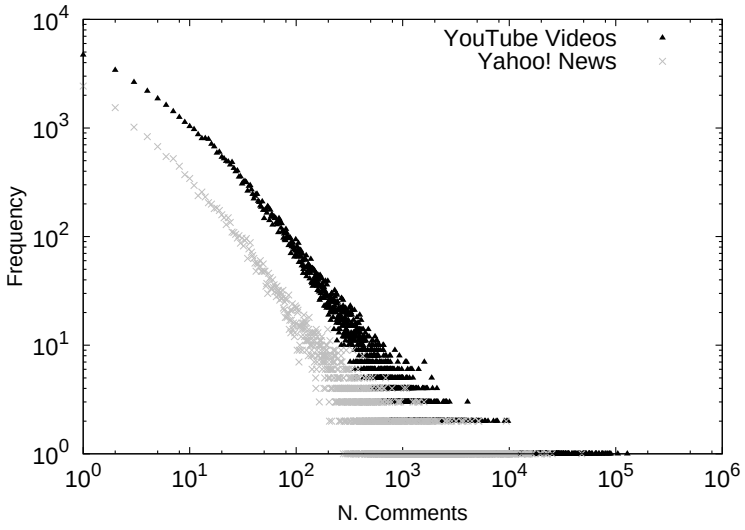


Figure 3.1: Distribution of number of comments for videos in YouTube and news stories in Yahoo! News.

where stopwords were removed and sentences were segmented using the GATE tool[4]. A closer inspection of the datasets revealed that users commenting on a news story tend to elaborate more on concepts and opinions, whereas users in YouTube often post very short comments that simply express favor or disfavor of video clips.

We also inspected the vocabulary of all comments for each dataset, and found that YouTube and Yahoo! News comments include 702,000 and 612,000 terms respectively. The overlap between lexicons is about 25% (165,000 terms). This lexical divide is mostly due to the different topics covered in the two datasets, but also caused by the commenting behavior being organically different in both sites, which is also suggested by the differences noted above (cf. Table 3.1). However, this should be interpreted cautiously, because both comment datasets include a high number of words with typos, abbreviations, etc.

On the other hand, there are also trends that are exhibited in both collections. Figure 3.1 shows the distribution of the number of comments per video and news story in the YouTube and Yahoo! News collections. The distributions follow the expected zipf-like pattern, characterized by having most of the energy contained within the first ranked elements and a subsequent long tail of additional low-represented elements [24, 51].

In Table 3.1, we also provide basic descriptive statistics about comment ratings. For the YouTube collection, we observe that ratings range from $-1,918$ to $4,170$ with a mean value of $\mu_r = 0.61$. For Yahoo! News, the ratings range from $-1,018$ to

[4] http://gate.ac.uk/

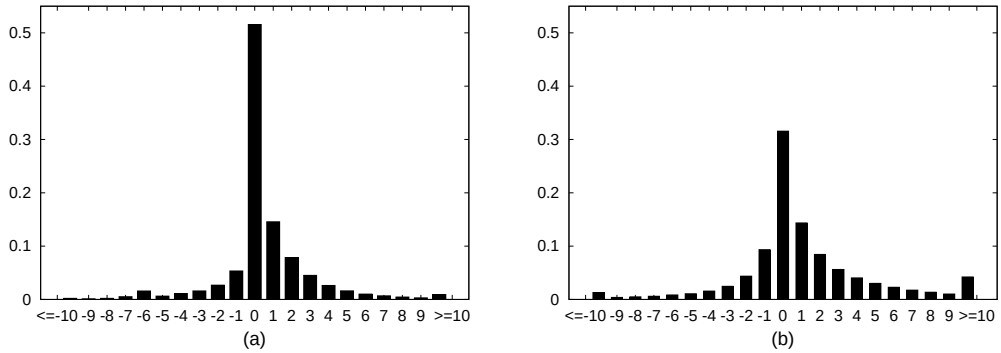


Figure 3.2: Distribution of comment ratings for (a) YouTube, and (b) Yahoo! News.

4,327 with a mean value of $\mu_r = 1.39$. While the minimum and maximum values for comment ratings in both datasets are in the same scale, on average, comments are rated higher in Yahoo! News than in YouTube. For a more detailed inspection, we show the distribution of comment ratings for both datasets in Figure 3.2. The following two main observations can be made. First, the distribution is asymmetric for positive and negative ratings, indicating that the community tends to cast more positive than negative votes. This behavior is more dominant in the Yahoo! News collection, as the mean rating score is almost twice as high as for video comments. Second, comments with rating 0 represent about 50% and 30% of the overall population for the YouTube and Yahoo! News collections, respectively, indicating that a substantial fraction of comments lack votes or are neutrally evaluated by the community.

3.3 Comment Ratings

3.3.1 Term Analysis of Rated Comments

The textual content of comments in Web 2.0 infrastructures can provide clues about their potential acceptance by the community. As an illustrative example we computed a ranked list of terms from a set of 100,000 comments with a rating of 5 or higher (high community acceptance) and another set of the same size containing comments with a rating of -5 or lower (low community acceptance). For stopwords removal and stemming of comments Lucene's SnowBallAnalyzer was used. For ranking the resulting terms, we used the Mutual Information (MI) measure [91, 144] from information theory which can be interpreted as a measure of how much the joint distribution of features X_i (terms in our case) deviate from a hypothetical distribution in which features and categories ("high community acceptance" and "low community acceptance") are independent from each other. Table 3.2 shows the top-50 (stemmed) terms extracted for each category. Note that some of the terms seem to emerge from the use of emoticons or similar

Table 3.2: Top-50 terms according to their MI values for accepted comments (with high comment ratings) vs. not accepted comments (with low comment ratings).

Terms for Accepted Comments

YouTube				Yahoo! News			
love	voic	gorgeous	scientolog	illeg	union	home	stay
song	hot	3	man	job	speech	rule	law
great	perfect	heart	talent	pay	hurrican	bless	holder
amaz	time	rocki	sweet	border	thing	dont	live
beauti	miss	john	inspir	polit	mexico	money	spend
awesom	feel	greatest	absolut	state	fire	time	cheney
cute	rock	cri	allison	friend	immigr	texa	storm
favorit	perform	scene	hill	sad	famili	hope	compani
music	nice	whitney	ador	work	need	go	dollar
lol	omg	brilliant	fantast	campaign	gun	new	busi
lt	jame	wonder	cool	govern	feder	taxpay	servic
xd	movi	luv		politician	citizen	crimin	
britney	sexi	part		doe	countri	vacat	

Terms for Unaccepted Comments

YouTube				Yahoo! News			
fuck	white	obama	cock	republican	racist	world	herman
suck	fat	de	comment	gop	christian	look	parti
gay	black	cunt	wtf	cain	vote	christ	presid
shit	fag	pussi	asshol	jesus	fact	nazi	israel
bitch	faggot	die	horribl	bagger	truth	white	gay
stupid	jew	bore	whore	wing	class	liber	teabagg
ass	retard	crap	Im	like	earth	black	american
nigger	kill	loser	lame	lol	obama	america	protest
ugli	fake	hell	racist	lie	bush	conserv	win
dont	idiot	peopl	hey	democrat	hate	right	kill
ur	dumb	shut	read	tea	rich	bibl	die
hate	bad	worst		jew	2012	zionist	
dick	guy	fuckin		god	fox	jewish	

symbols. For instance, the sequence "<3" is used to represent a heart symbol. On the other hand, "de" might be part of a URL (".de" for Germany).

Obviously many of the "accepted" comments in the YouTube collection contain terms expressing sympathy or commendation (*love*, *fantast*, *greatest*, *perfect*). "Unaccepted" comments, on the other hand, often contain swear words (*retard*, *idiot*) and negative adjectives (*ugli*, *dumb*); this indicates that offensive comments are, in general, not promoted by the community. We applied the same term analysis procedure on the Yahoo! News collection. The difference between terms from the "accepted" and "unaccepted" categories is still visible but not as significant as for YouTube. Yahoo! News is more sensitive to the language used by users and it enforces stricter policies concerning insults and hate phrases [5], which makes the content in accepted and unaccepted comments lexically more similar.

Table 3.3 shows a couple of hand-picked comments from the YouTube and Yahoo! News sets to illustrate both accepted and unaccepted comments. For instance, those comments that are supporting Charlie Brown, a sympathetic and full of hope child-

[5] http://help.yahoo.com/kb/index?locale=en_US&page=content&y=PROD_NEWS&id=SLN2292

Table 3.3: Examples of comments belonging to the categories *"accepted"* and *"unaccepted"*.

Rating	Text
	Accepted Comments
	YouTube
17	this is true rock and roLL!! i feel good!! (^^)
12	micheal should have lived longer relly.
11	he has one of the most beautiful voices i have heard. He is a sweetheart <3
7	Poor Charlie Brown. I just wanna jump in there and give him a big hug and tell him its alright!! =(
	Yahoo! News
13	I wonder if Bauchmann understands she has the same chance of becoming President as I do?
12	The Government needs to get out of the way and let the markets and economy heal themselves. Washington knows nothing about making money - only how to spend it !
11	this is very good news for my wife who suffers from ALS
10	Great that some were found alive.Sorry about the ones that did not make it.
	Unaccepted Comments
	YouTube
-13	the only reason they made rocky balboe was beacause the whites were jealous of muhammed ali
-12	this song is so stupid..! this song should go to hell maybe its a good song to get high to tho?
-7	British accents are annoying in films for some reason, no offence to anyone..
	Yahoo! News
-15	America is the world every other so called country is nothing but animals
-15	Hurricane Hype or Irene was nothing. I seen more rain and wind from spring thunderstorms.
-11	White people don't commit crimes they just murder their babies, parents and siblings

character from a popular American series that fails in everything he does are liked by the YouTube community. In contrast, a rude criticism about races and nations is disliked by the community, as well as being very negative towards a popular song. In the case of Yahoo! News, a comment predicting the low chances of Michele Bachmann of becoming president, or comments expressing support for someone suffering from amyotrophic lateral sclerosis (ALS) or for people suffering after earthquakes are highly accepted by the community. Similar to YouTube, being racist and criticizing countries without arguments will likely trigger negative comment ratings from the Yahoo! News readers.

3.3.2 Sentiment Analysis of Rated Comments

Does language and sentiment used by the community have an influence on comment ratings?

In this section, we make use of the publicly available SentiWordNet thesaurus to study the connection between the sentiment features of comments and the ratings they get. We aim at understanding how *the way users express opinions* affects comment approval from the rest of the community, regardless of the actual opinion stated.

We now describe our statistical comparison of the influence of sentiment scores in comment ratings. In our experiments, we assigned a sentivalue to each comment by computing the averages for *pos* and *neg* over all words in the comment that have an entry in SentiWordNet, a sentiment dictionary which we described in Section 2.2. We restrict our analysis to adjectives, as we observed the highest accuracy in SentiWordNet for these.

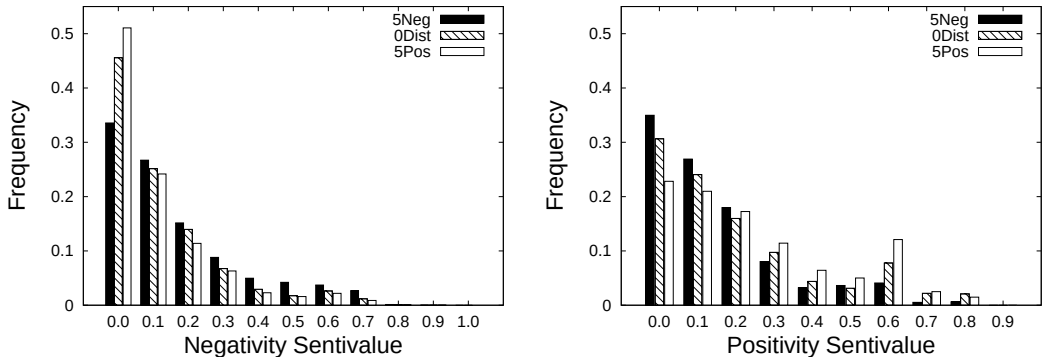
Our intuition is that the choice of terms used to compose a comment may provoke strong reactions of approval or denial in the community, and therefore determine the final rating score. For instance, comments with a high proportion of offensive terms would tend to receive more negative ratings. We used comment-wise sentivalues, computed as explained above, to study the presence of sentiments in comments according to their rating.

To this end, we first subdivided the data set into three disjoint partitions:

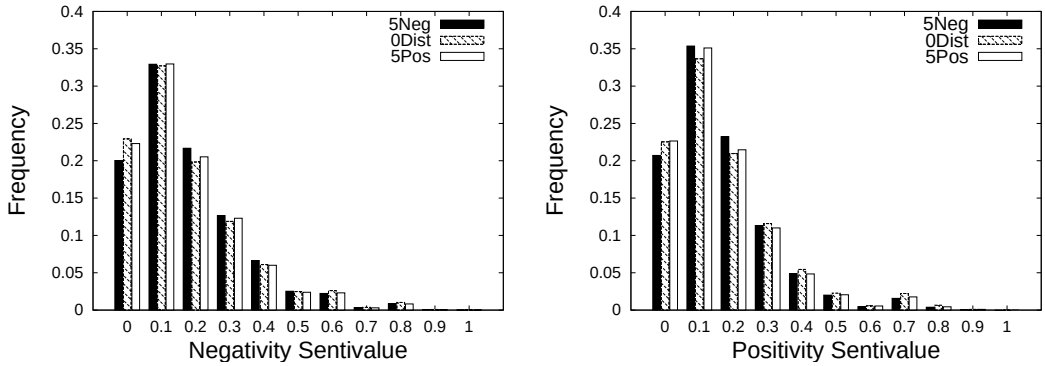
- **5Neg**: The set of comments with rating score r less or equal to -5, $r \leq -5$.
- **0Dist**: The set of comments with rating score equal to 0, $r = 0$.
- **5Pos**: The set of comments with rating score greater or equal to 5, $r \geq 5$.

We then analyzed the dependent sentiment variables "positivity" and "negativity" for each different partition. Detailed comparison histograms for these sentiments are shown in Figure 3.3. This figure shows a clearly different behavior in YouTube and Yahoo! News. In the case of YouTube, the results follow our intuition: negatively rated comments (**5Neg**) tend to contain more negative sentiment terms than positively rated comments (**5Pos**). This is reflected by a lower frequency of sentivalues at negativity level 0.0 along with consistently higher frequencies at negativity levels ≥ 0.1. Similarly, positively rated comments tend to contain more positive sentiment terms. We also observe that comments with rating score equal to 0 (**0Dist**) have sentivalues in between, in consonance with the initial intuition.

In the case of the Yahoo! News dataset, the observed pattern is substantially different. The distribution of sentivalues, both positive and negative, does not have a clear



(a) YouTube

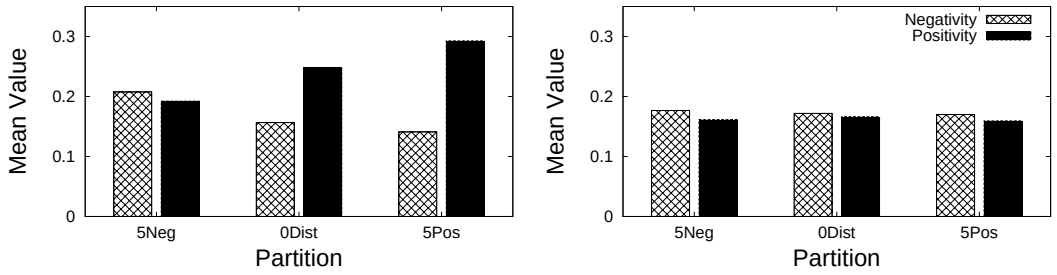


(b) Yahoo! News

Figure 3.3: Distribution of comment negativity, and positivity for (a) YouTube and (b) Yahoo! News.

dependency with respect to the considered partition. That is, comment ratings are not as influenced by the sentiment orientation of the words contained in them. This result is in consonance with the observations from Section 3.3.1. Comments in Yahoo! News are subject to stricter policies which reduces the occurrence of offensive terms. In addition, it is expected that well written comments could attract negative ratings just because of the diversity of opinions in the matters normally covered in the news. This results in a lower dependency of ratings with respect to the sentiment orientation of comments.

We conducted tests to examine the statistical difference of the average senti-values (both positive and negative distributions) across the three groups defined (**5Neg**, **0Dist**, **5Pos**) in both datasets. To this end, we selected a random sample of $5,000$ comments for which sentiment values were available in SentiWordNet. The analysis of variance (ANOVA) test for each of these 4 conditions (2 sentiment orientations $\times$ 2 datasets) systematically resulted in a strong support of the significance of the difference (p-value < 0.01) across the three groups. Figure 3.4 shows the difference of mean values for negativity and positivity, revealing that negative sentivalues are predominant in



(a) YouTube (b) Yahoo! News

Figure 3.4: Comparison of mean senti-values for comments with different kinds of community ratings in (a) YouTube and (b) Yahoo! News.

negatively rated comments, whereas positive sentivalues are predominant in positively rated comments. This difference, however, is clearly more noticeable in the YouTube dataset in consonance with previously reported results.

3.3.3 Temporal Characteristics of Comment Ratings

What is the average "lifetime" of a comment in terms of the community feedback it attracts? Do ratings in the earlier lifetime of a comment affect subsequent ratings? Are there preferential attachment effects, i.e. do positive/negative ratings at an early stage lead to a bias towards even more positive/negative ratings?

In order to study the temporal dynamics of comment rating and reply behavior, on the 15th of February 2013 we gathered the stories from Yahoo! News published on that day (amounting to a total of 187 news stories). For these stories, we crawled all of the available comments for a 7 day time interval, updating content of new comments and information on the temporal development of comment ratings iteratively in a round-robin fashion over the news stories. We chose a 7 day interval for the crawl as we noticed that most of the stories were removed from the system after one week. This resulted in a set of 18,902 comments being updated up to 59 times within the one week period using our data gathering strategy.

Figure 3.5 sketches the crawling strategy for an individual comment starting with the posting time of the comment. For each comment we defined a set of fixed time points for updating comment rating information, corresponding to time periods of 2h (the warm-up interval), 4h, 8h, 16h, 1 day, 2 days, 3 days, and 4 days after the posting time of the comment. The red vertical lines depict the actual time when comment information was crawled and updated. Updates within a range of 20% of the fixed time periods (indicated by blue circles) were assigned to the corresponding fixed time point. This estimate was necessary due to the lack of timestamps for updates and

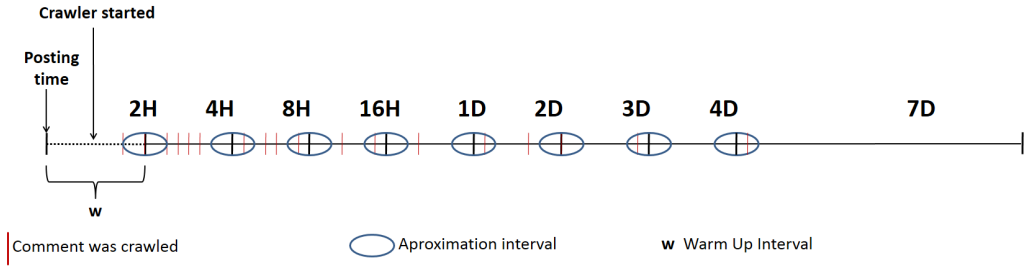


Figure 3.5: Crawling strategy for temporal analysis.

the availability of just very rough approximations for posting times. Note that due to the difficulties implied in crawling information on posted and updated comments and limitations in the possible number of http requests per hour, time latencies occur which can result in missing some of the comments in their early life (warm-up interval) but also for some other points in time. Therefore, our final analysis was conducted on a subsample of comments fulfilling the following two criteria: 1) they were crawled in their early life (we experimentally chose a warm up interval of 2 hours), and, 2) the crawler gathered the comment rating information for the fixed time points as defined above. The final dataset used for our analysis consisted of 2,404 comments for 62 news stories.

Figure 3.6 shows the temporal development of the average number of likes, dislikes, ratings, and replies for comments. We observe that majority of the user interactions occur within the first 8 hours after a comment is posted. About 1 day after the posting there are clear saturation effects, indicating that comments do not receive much feedback from the community after that time period.

In order to examine the influence of early ratings, we split the dataset of comments

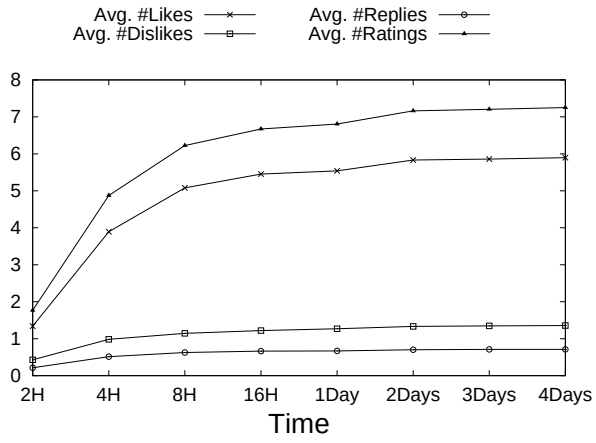


Figure 3.6: Temporal evolution of average number of comment ratings, likes, and dislikes; and average number of replies.

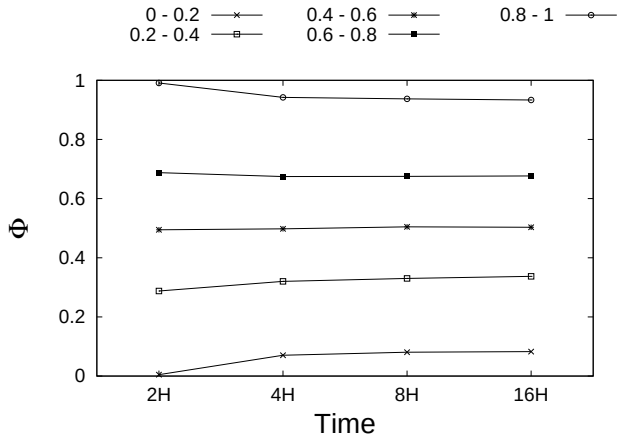


Figure 3.7: Temporal evolution of average acceptance ratios for different ranges of values for initial comment acceptance ratios Φ.

obtained through the time-aware crawl into five disjoint subsets depending on their ratio Φ of likes and overall ratings in the warm-up phase (i.e. the first 2h after their posting time). To this end, we split the range of possible values of Φ ([0,1]) into 5 equidistant subintervals and assigned the comments that had obtained at least one rating in the warm-up phase to the corresponding subset.

Figure 3.7 shows the temporal development of the average Φ values for different initial ranges of Φ. We observe that for the different starting values, the ratio Φ of likes and overall ratings stays almost constant, indicating that this ratio remains stable and that there are no preferential attachment effects.

3.3.4 Comment Ratings and Polarizing Content

So far we mainly focused on the comments themselves and did not consider the associated shared content. In this section, we will study the relationship between comment ratings and polarizing content, more specifically tags/topics and videos. By "polarizing content" we mean content likely to trigger diverse opinions and sentiment, examples being content related to the war in Iraq or the presidential election in contrast to rather "neutral" topics such as chemistry or physics. Intuitively, we expect a correspondence between diverging and intensive comment rating behavior and polarizing content in YouTube.

Variance of Comment Ratings as Indicator for Polarizing Videos In order to identify polarizing videos, we computed the variance of comment ratings for each video in our dataset. Figure 3.8 shows examples of videos with high versus low rating variance (in our specific examples videos about an Iraqi girl stoned to death, Obama, and protest on Tiananmen Square in contrast to videos about The Beatles, cartoons,

Figure 3.8: Videos with high (upper row) versus low variance (lower row) of comment ratings.

and amateur music). In order to show the relation between comment ratings and polarizing videos more systematically , we conducted a user evaluation of the top- and bottom-50 videos sorted by their variance. These 100 videos were put into random order, and evaluated by 5 users on a 3-point Likert scale (3: polarizing, 1: rather neutral, 2: in between). Participants consisted of PhD students and PostDocs from the institution of the first author of our published paper [120]. The assessments of the different users were averaged for each video, and we computed the inter-rater agreement using the κ-measure [112], a statistical measure of agreement between individuals for qualitative ratings. The mean user rating for videos on top of the list was 2.085 in contrast to a mean of 1.25 for videos on the bottom (inter-rater agreement $\kappa = 0.42$); this is quite a high difference on a scale from 1 to 3, and supports our hypothesis that polarizing videos tend to trigger more diverse comment rating behavior. A t-test confirmed the statistical significance of this result (t= 7.35, d.f. $= 63$, P < 0.000001).

Variance of Comment Ratings as Indicator for Polarizing Topics We further studied the connection between comment ratings and video tags corresponding to polarizing topics. To this end we selected all tags from our dataset occurring in at least 50 videos resulting in $1,413$ tags. For each tag we then computed the average variance of comment ratings over all videos labeled with this tag. Table 3.4 shows the top- and bottom-25 tags according to the average variance. We can clearly observe a higher tendency for tags of videos with higher variance to be associated with more polarizing topics such as *presidential*, *islam*, *irak*, or *hamas*, whereas tags of videos with low variance correspond to rather neutral topics such as *butter*, *daylight* or *snowboard*. There

Table 3.4: Top and Bottom-25 tags according to the variance of comment ratings for the corresponding videos.

High comment rating variance				
presidential	nomination	muslim	shakira	islam
campaign	station	itunes	grassroots	nice
xbox	barack	efron	zac	iraq
3g	kiss	obama	deals	celebrities
jew	space	shark	hamas	kiedis
Low comment rating variance				
betting	turns	puckett	tmx	tropical
skybus	peanut	defender	f-18	vlog
butter	chanukah	form	savings	iditarod
lent	daylight	egan	snowboard	havanese
menorah	casserole	1040a	1040ez	booklet

are also less obvious cases an example being the tag *xbox* with high rating variance which is due to polarizing gaming communities strongly favoring either Xbox or other consoles such as PS3, another example being *f-18* with low rating variance, a fighter jet that is discussed under rather technical aspects in YouTube (rather than in the context of wars). We evaluated this tendency in a user experiment with 3 assessors. Again, participants consisted of PhD students and PostDocs from the affiliation of the first author of our published paper [120]. We followed the same strategy as previously described, using a 3-point Likert scale and presenting the tags to the assessors in random order. The mean user rating for tags in the top-100 of the list was 1.53 in contrast to a mean of 1.16 for tags on the bottom-100 (with inter-rater agreement $\kappa = 0.431$), supporting our hypothesis that tags corresponding to polarizing topics tend to be connected to more diverse comment rating behavior. The statistical significance of this result was confirmed by a t-test (t=4.86, d.f. = 132, P = 0.0000016).

In order to study topics beyond individual tags and to obtain more context-related information, we additionally employed Latent Dirichlet Allocation (LDA) [15] and modeled each tag-based representation of a video as a mixture of latent topics. For performing the topic modelling we used the LDA implementation in the Mallet library. [6] In a nutshell, given a set of term sets (videos v_i represented by their tags in our case) and the desired number of latent topics, k, LDA outputs the probabilities $P(z_j|v_i)$ that topic z_j is contained in video v_i. In addition, LDA computes term probabilities $P(t_j|z_i)$ for tags t_j; the terms with the highest probabilities for a latent topic z_i can be used to represent that topic.

We empirically chose the number of latent topics as 200 for our YouTube dataset.

[6] http://mallet.cs.umass.edu/

Table 3.5: Most probable terms for the top-5 and bottom-5 latent topics according to the comment rating variance of the corresponding videos.

Top-5 topics connected to videos with high comment rating variance				
TOPIC 1:	TOPIC 2:	TOPIC 3:	TOPIC 4:	TOPIC 5:
saddam	vegas	shakira	de	clinton
hussein	simpson	lie	mayo	obama
iraq	las	hips	cinco	barack
pacing	oj	don	san	bill
dr	peanut	dont	mexico	hillary
pluto	butter	wolf	mexican	president
hanging	recall	hurley	diego	john
stc	prison	elizabeth	jose	election
division	trial	dance	california	bush
healthcare	talent	live	latino	politics
Bottom-5 topics connected to videos with low comment rating variance				
TOPIC 1:	TOPIC 2:	TOPIC 3:	TOPIC 4:	TOPIC 5:
kurt	tax	easter	dance	carey
vonnegut	taxes	chanukah	girl	mariah
language	income	bunny	hot	mary
arts	irs	egg	blonde	ron
book	aid	menorah	katie	lol
science	form	gas	babe	porn
theatre	free	jewish	big	funny
learn	forms	prices	ass	cannon
humanities	video	eggs	cindy	fail
art	federal	darfur	black	awesome

Analogously to our method for identifying terms related to polarizing topics, we computed the average variance of comment ratings over all videos that belong to a latent topic, weighting the contributions of the videos by their probability values $P(z_j|v_i)$. Table 3.5 shows the top-5 and bottom-5 latent topics (represented by their most probable terms) according to their average variance scores. Similar as for individual terms, polarizing and neutral topics can be successfully distinguished. In particular, the topics that are centered around *Iraq*, *O.J. Simpson trial* and *American politics* (i.e, the first, second and last columns of top-5 topics in Table 3.5, respectively) are obviously polarizing. On the other hand, the bottom-5 topics look rather neutral, being related to the issues like *Kurt Vonnegut* (an American writer), *tax forms*, *girls*, etc.

3.3.5 Predicting Comment Ratings

Can we predict community acceptance? Our term- and SentiWordNet-based analyses described in the previous sections indicate the discriminative character of terms occurring in comments with respect to comment ratings.

We used machine learning and term-based representations of comments to automatically categorize comments as likely to obtain a high overall rating or not. In order to classify comments into categories "accepted by the community" or "not accepted", we employed linear support vector machines (SVMs) as they have been shown to perform well for various classification tasks (see, e.g., [50, 80]). Comments labeled as "accepted" or "not accepted" are used to train the classification model. The feature vector of a comment was constructed using the the frequencies of the terms occurring in the comment, normalized by the number of terms in the comment. We removed stopwords and terms occurring just once in the corpus, and applied Lucene's SnowBallAnalyzer for stemming. We used the LIBSVM [26] implementation of support vector machines using a linear kernel and cost parameter C=0.1.

How can we obtain sufficiently large training sets of "accepted" or "not accepted" comments? We are aware that the concept is highly subjective and problematic. However, the amount of community feedback in YouTube results in large annotated comment sets which can help to average out noise in various forms and, thus, reflects to a certain degree the "democratic" view of a community. To this end we considered distinct thresholds for the minimum comment rating for comments. Formally, we obtain a set $\{(\vec{c_1}, l_1), \ldots (\vec{c_n}, l_n)\}$ of comment vectors $\vec{c_i}$ labeled by l_i with $l_i = 1$ if the rating lies above a threshold ("positive" examples), $l_i = -1$ if the rating is below a certain threshold ("negative" examples).

Setup We performed different series of binary classification experiments of YouTube comments into the classes "accepted" and "not accepted" as introduced before. For our classification experiments, we considered different levels of restrictiveness for these classes. Specifically, we considered distinct thresholds for the minimum and maximum ratings (above/below +2/-2, +5/-5 and +7/-7) for comments to be considered as "accepted" or "not accepted" by the community.

We also considered different amounts of randomly chosen "accepted" training comments (T = 10,000, 50,000, 200,000) as positive examples and the same amount of randomly chosen "unaccepted" comments as negative samples (where that number of training comments and at least 1,000 test comments were available for each of the two classes). For testing the models based on these training sets we used the disjoint sets of remaining "accepted" comments with same minimum rating and a randomly selected disjoint subset of negative samples of the same size. We performed a similar experiment by considering "unaccepted" comments as positive and "accepted" ones as negative, thus, testing the recognition of "bad" comments. We also considered the scenario of discriminating comments with a high absolute rating (either positive or

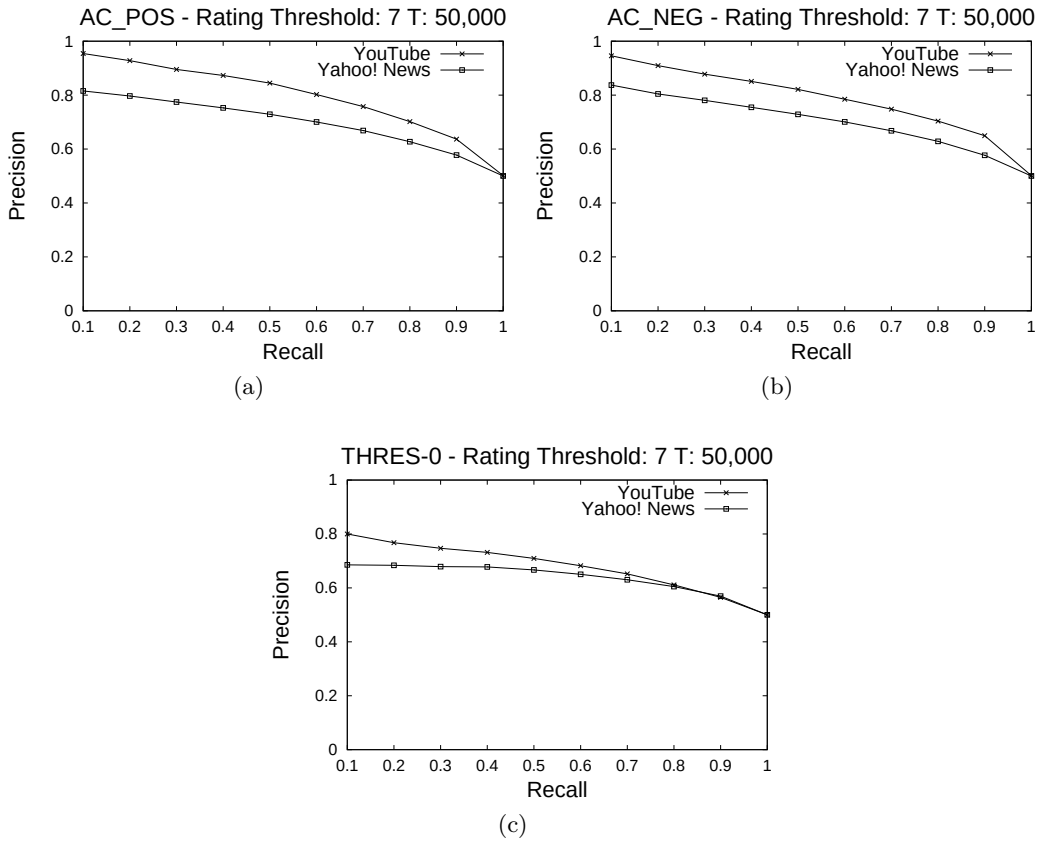


Figure 3.9: Precision-recall curves for comment rating prediction.

negative) against unrated comments (rating = 0). The three scenarios are labeled **AC_POS**, **AC_NEG**, and **THRES-0** respectively.

Results Our quality measures are the precision-recall curves as well as the precision-recall break-even points (BEPs) for these curves (i.e. precision/recall at the point where precision equals recall, which is also equal to the F1 measure, the harmonic mean of precision and recall in that case). The results for the BEP values are shown in Table 3.6. The detailed precision-recall curves for the example case of T=50,000 training comments class and thresholds +7/-7 for "accepted"/ "unaccepted" comments are shown in Figure 3.9 for YouTube and Yahoo! News.

The main observations are:

- All three types of classifiers provide good performance for the YouTube dataset. For instance, the configuration with T=50,000 positive/negative training comments and thresholds +7/-7 for the scenario **AC_POS** leads to a BEP of 0.738. Consistently, similar observations can be made for all examined configurations.

- Trading recall against precision for YouTube leads to applicable results. For

Table 3.6: Comment rating classification: BEPs for different training set sizes T and different rating thresholds.

	YouTube			Yahoo! News		
T	Rating ≥ 2	Rating ≥ 5	Rating ≥ 7	Rating ≥ 2	Rating ≥ 5	Rating ≥ 7
			AC_POS			
10,000	0.659	0.696	0.721	0.577	0.624	0.649
50,000	0.679	0.715	**0.738**	0.586	0.654	**0.676**
200,000	0.693	-	-	0.597	0.668	-
			AC_NEG			
10,000	0.659	0.693	0.721	0.574	0.628	0.646
50,000	0.678	0.714	**0.734**	0.588	0.652	**0.676**
200,000	0.691	-	-	0.604	0.668	-
			THRES-0			
10,000	0.595	0.620	0.640	0.566	0.609	0.620
50,000	0.605	0.642	0.663	0.577	0.628	0.642
200,000	0.621	0.656	**0.671**	0.618	0.642	**0.656**

instance, we obtain prec=0.872 for recall=0.4, and prec=0.954 for recall=0.1 for **AC_POS**; this is useful for finding candidates for interesting comments in large comment sets.

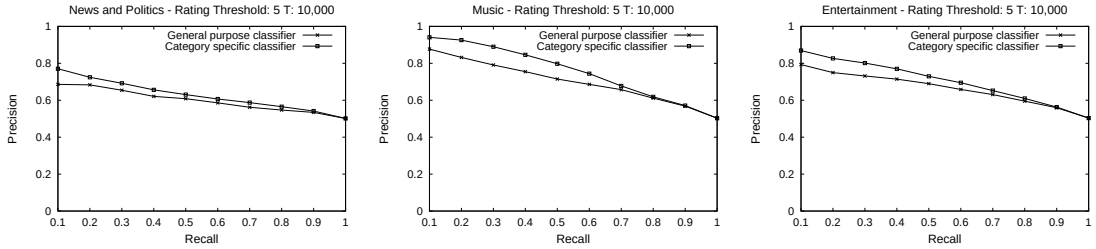
- Classifiers perform worse for Yahoo! News; with T=50,000 positive/negative training comments and thresholds +7/-7 we obtain a BEP of 0.676 for predicting positively rated comments. This is expected as our discriminate term and sentiment analyses described in previous sections revealed less clear patterns for that dataset. However, trading precision for recall can help again to obtain more applicable results (prec=0.815 for recall=0.1).

- Classification results tend to improve, as expected, with increasing number of training comments. Furthermore, classification performance increases with higher thresholds for community ratings for which a comment is considered as "accepted".

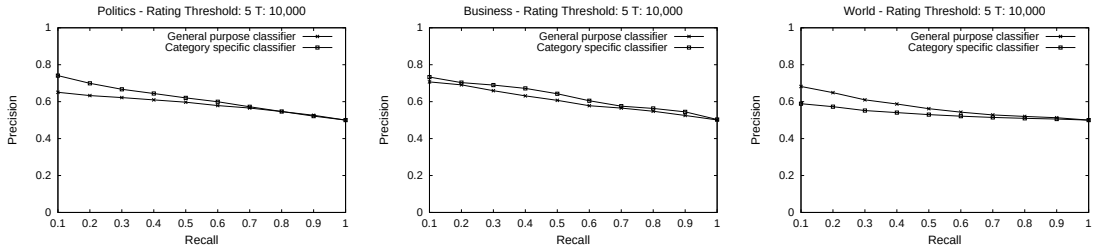
Overall our results confirm our intuition that there exist discriminative terms which depend on the comment ratings (see Table 3.2 in Section 3.3.1) and which enable us to train meaningful classification models.

3.3.6 Category-Specific Rating Prediction

Does video content, especially the category of the video, have an influence on comment rating behavior, and can this information be leveraged to improve classification performance? To answer this research question we conducted category specific classification experiments, using the same category for testing and training. We compared



(a) YouTube



(b) Yahoo! News

Figure 3.10: Precision-recall curves for category-specific rating prediction experiments.

the performance of these classifiers with "general purpose" classifiers using training sets randomly chosen across all categories as in the experiments described above. Both YouTube and Yahoo! News provide category information for their videos.

For YouTube we selected videos from the categories "News and Politics","Music" and "Entertainment"; for Yahoo! News we selected "Politics", "Business" and "World". For each experiment we randomly chose 10,000 "accepted" and the same number of "unaccepted" documents using a threshold of +5/-5 respectively. For each of the described categories from Yahoo! News and YouTube we selected the test comments to be from the same category as the training comments. We then compared the performance with that of a classifier trained on comments randomly chosen across categories. Feature vector construction, machine learning algorithm, and parameter settings were the same as in previous experiments described in this section.

Figure 3.10 shows the resulting precision-recall curves. We observe that the category-specific classifiers consistently outperform the "general purpose" classifiers for all tested categories both for YouTube and Yahoo! News. For instance, for the "Entertainment" category in YouTube the performance boost in the recall range of 0.1 up to 0.4 is more than 5%. Note that, in order to obtain consistent numbers of training comments on a per category level we had to restrict trainings set sizes in this experiment, resulting in a decrease of absolute performance values in comparison to our previous experiments.

3.4 Discussion Threads and Replies

Besides comment ratings, another important community feedback mechanism in many Social Web environments is the option of posting replies to comments. In this thesis, we refer to comments that received at least one reply as *seed comments*. Likewise, we refer to comments that are *not* replies to other comments as *main comments*. In the remainder of this section we compare the distribution of replies for YouTube and Yahoo! News, study the relationship between comment ratings and replies, and apply machine learning to identify comments likely to receive replies. Predicting and promoting comments that are likely to trigger an online discussion can help to increase user participation and engagement within online collaborative platforms.

3.4.1 Analysis of Replies and Ratings

How prominently is the reply feature used, and what is that connection between discussion threads and ratings?

Replies in YouTube and Yahoo! News Table 3.7 shows a couple of hand-picked examples of comments that received replies from our YouTube and Yahoo! News datasets. Comments elaborating on pros and cons of the Xbox 360 gamebox, or discussing whether Jimi Hendrix was as good as Stevie Ray Vaughan received 11 and 7 reply responses respectively. In Yahoo! News, comments discussing the danger of abusing alcohol in comparison to using marijuana, or on the usefulness of producing high quality goods in foreign countries received 23 and 11 replies respectively.
Table 3.8 shows the basic frequency statistics for comments that received one or more

Table 3.7: Examples of comments belonging to the category *"seed comments"* (comments that received replies).

Nr. Replies	Text
	YouTube
11	I like Xbox 360 better then PS3 BUT..this is a huge BUT the games I like best on xbox is soo much money. Most xbox games are abut 70 bucks but some can be even higher.
7	I agree guitarboii101 that jimi hendrix was revolutionary, but..if you say who is the BEST guitar player ever.. it has to be stevie ray vaughan.
7	Congratulations to Agassi on a great career. He did one thing that Roger Federer may never do: win all 4 grand slams.
	Yahoo! News
23	you can drink all you want ,but you can't smoke a joint! never seen anybody die from smoking to much weed ,a dead sleep yes, but how many people die from drinking to much!!
11	Why is everyone complaining about buying Chinese when alot of you have iPhones that are made in China.
7	NASA conveniently comes up with "discoveries" when their funding is out to be lost.

Table 3.8: Basic statistics for seed and reply comments in the YouTube and Yahoo! News corpora.

	YouTube			Yahoo! News		
	Unreplied	Seeds	Replies	Unreplied	Seeds	Replies
Amount	3,470,413 (56.4 %)	827,603 (13.5 %)	1,851,849 (30.1 %)	1,599,228 (29.2%)	1,139,833 (20.9 %)	2,739,426 (49.9 %)
Avg. #words	7.03	11.11	9.69	16.50	21.52	12.75
Median #words	4	6	6	10	13	8
Stdev. #words	8	9.84	10.43	24.54	29.06	17.74
Avg. #sentences	1.68	2.12	2.01	3.06	3.39	2.47
Median #sentences	1	1	1	2	2	2
Avg. rating	0.90	0.27	0.19	2.24	2.60	0.38
Median rating	0	0	0	1	1	0
Stdev. rating	9.04	7.83	7.38	4.92	22.81	2.59
Min. rating	-710	-1,918	-445	-66	-1,018	-210
Max. rating	3,807	2,693	4,170	722	4,327	238

replies in YouTube and Yahoo! News as well as statistics for the corresponding thread sizes. We observe that Yahoo! News contains a higher proportion of seed comments among main comments than YouTube. Furthermore, almost 50% of all comments in our Yahoo! News collection are replies to other comments, in comparison to just 30% for the YouTube collection (in consonance with the 23.4% reported in [128] for a more recent sample of YouTube). We also observe that both seed and reply comments tend to be longer (i.e. contain more sentences and words) in Yahoo! News. These results indicate that Yahoo! News users are more likely to engage in discussions, triggered by the specific characteristics of news stories (e.g. novelty, controversy).

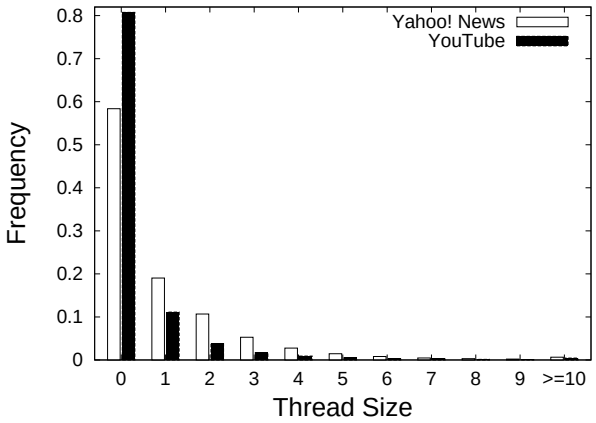


Figure 3.11: Distribution of thread sizes (number of replies) for the YouTube and Yahoo! News corpora.

Figure 3.11 shows the distribution of the number of comments without replies and seed comments with respect to the number of replies (we define comments without replies as threads with size 0). Similar to the larger number of comment ratings al-

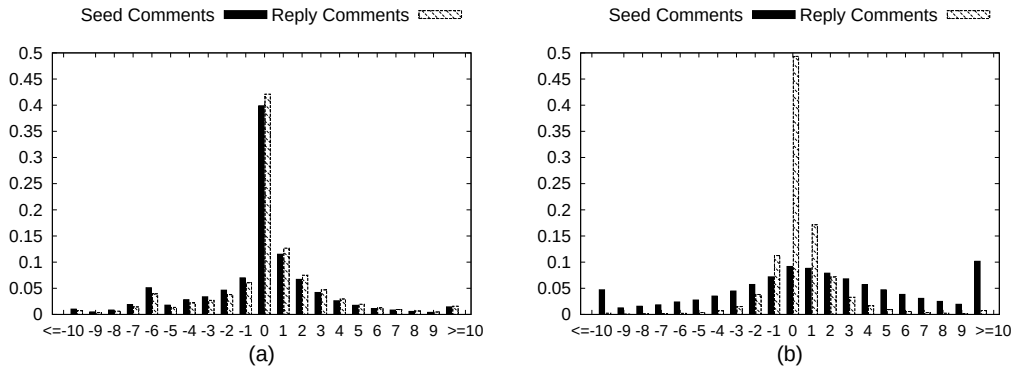


Figure 3.12: Distribution of ratings for seed and reply comments in (a) the YouTube dataset and (b) the Yahoo! News dataset.

ready observed for the Yahoo! News collection (cf. Figure 3.2), the number of replied comments in that collection is higher at all levels of thread size. We might expect that this is due to the wider variety of content in YouTube in comparison to Yahoo! News where political topics are predominant. However, an analysis of reply behavior in YouTube restricted to videos from the category *news and politics* did not reveal any significant differences with respect to the results found for the complete YouTube collection, hinting at a homogeneous reply behavior across categories in this community. This result provides additional support that the media types (i.e. videos vs. news stories) play a central role and that the particularities of users in each community are the main responsible for these different usage patterns.

Threads and Ratings Figure 3.12 shows the distribution of the average rating of seed comments and replies for both datasets. The distribution of seed comments for Yahoo! News shows longer tails for both positively and negatively rated comments, whereas ratings in YouTube concentrate around zero, following the trend shown in Figure 3.2. An interesting artifact to be noticed in this figure is the peak exhibited by *replies* with a rating value of zero in Yahoo! News. The most likely explanation that we found for this behavior is that the Yahoo! News web interface does not show comment replies by default. An explicit user action (clicking on a link below the seed comment) is required so replies are shown and can be rated. As a result, users are less likely to see replies and rate them.

We also studied the dependency of ratings for seed comments and the length of the corresponding discussion threads. In Figure 3.13 we see that for Yahoo! News the average rating of seed comments grows with increasing thread size. An explanation for this is that comments initiating larger threads draw more attention. Note that in general the distribution of comment ratings is skewed towards positivity in Yahoo!

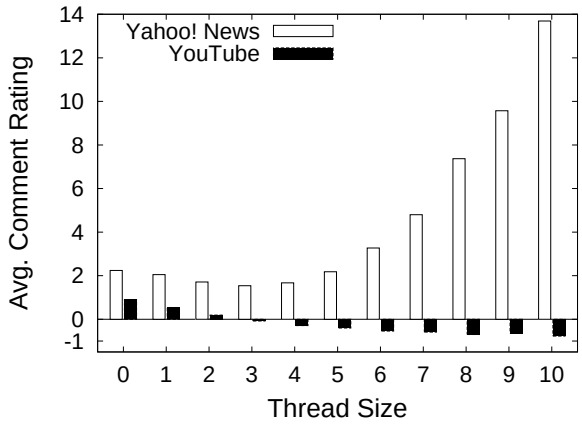


Figure 3.13: Average comment rating of seed comments in Yahoo! News and YouTube with respect to thread size.

News (cf. Figure 3.12). Interestingly, for YouTube we observe a decrease of ratings, and even a trend towards negative ratings, with increasing thread size; manual inspection (conducted by the second author of our published paper [121]) of a random sample of 300 threads containing 20 or more comments confirmed that longer discussions in YouTube often tend to be rude or "flame wars" (82.6% of the threads in our sample with a 95% confidence interval of ±4%). More specifically, we refer to flame wars as discussions/interactions characterized by "words of profanity, obscenity, and insults that inflict harm to a person or an organization" as discussed in literature in Social Sciences [6].

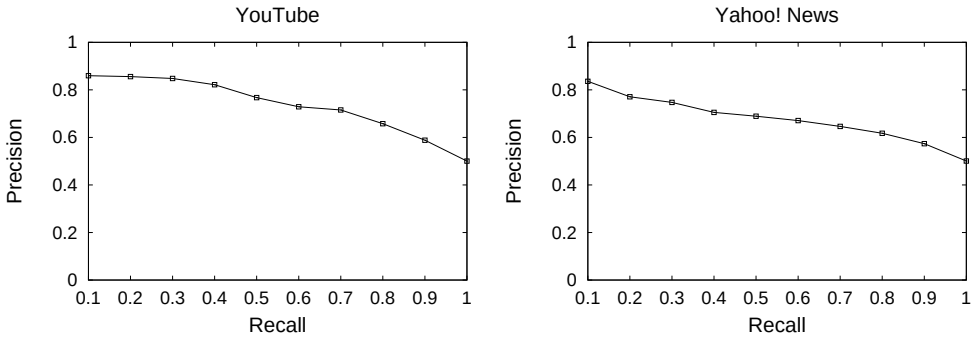
Note that, so far, we only considered the overall rating of the comments. The Yahoo! News data comprises additional information on how ratings of individual comments decompose into positive and negative votes ("likes" and "dislikes"). Section 3.5 will be dedicated to this topic and we will revisit discussion threads in that context.

3.4.2 Predicting the Responsivity on Comments

In order to study if information obtained from the textual content of comments can be used to predict seed comments (i.e., comments that receive replies and start a discussion thread), we performed different series of binary classifications of YouTube and Yahoo! News comments into the classes "Seed" and "Unreplied". Here we considered different levels of restrictiveness for these classes. Specifically, we considered distinct thresholds R for the minimum number of received replies (2,5,7, and 9) for comments to be considered as "Seeds"; comments with no replies were considered as "Unreplied". Our rationale for studying different reply thresholds was to explore how the amount of replies can influence the classifier performance. We also considered different amounts

Table 3.9: BEPs for classification of seed comments vs. comments without replies.

T	YouTube				Yahoo! News			
	R≥2	R≥5	R ≥ 7	R ≥ 9	R≥2	R≥5	R ≥ 7	R ≥ 9
5,000	0.617	0.636	0.681	0.692	0.567	0.620	0.632	0.636
10,000	0.635	0.654	0.690	0.712	0.579	0.621	0.644	0.655
30,000	0.637	0.678	-	-	0.588	0.633	-	-
100,000	0.649	-	-	-	0.591	-	-	-



(a) (b)

Figure 3.14: Precision-recall curves for predicting replied comments for (a) the YouTube and (b) the Yahoo! News dataset.

of randomly chosen "Seed" training comments (T = 5,000, 10,000, 30,000, 100,000) as positive examples and the same amount of randomly chosen "Unreplied" comments as negative samples (where that number of training comments were available for each of the two classes). We tested the models on the disjoint sets of remaining seed comments (with different thresholds R for the minimum number of received replies) and a randomly selected disjoint subset of "unreplied" comments of the same size. The sizes of the test sets varied for YouTube from 26,170 (T=10,000, R ≥ 9) to 680,726 (T=5,000, R ≥ 2) and for Yahoo! News from 26,080 (T=10,000, R ≥ 10k) to 1,226,248 (T=5,000, R ≥ 2). We used LIBSVM with linear kernel and default parameterization. Feature vectors for comments were constructed as described in Section 3.3.5.

The resulting values of the precision-recall break-even-point (BEP) are shown in Table 3.9. We observe that the performance improves with increasing reply threshold R; this is expected as comments with more replies are more prominent representatives of comments triggering discussions. Classification performances are comparable for both Social Web environments studied.

The detailed precision-recall curves for the best-performing setting (i.e. $R \geq 9$) are shown in Figure 3.14. Due to the difficulty of the task, reply prediction is not feasible for high-recall scenarios. However, trading recall against precision leads to

applicable results. For instance, given a recall of 0.1, we obtain a precision of 0.859 for the YouTube dataset and a precision of 0.836 for the Yahoo! News data set. This is useful for automatically identifying at least part of the most likely candidates for comments triggering discussions, and can still help to mine a substantial amount of interesting seed comments from large datasets.

3.5 Controversial Comments

So far we have explored comment ratings as a single aggregate value for the community acceptance of a given comment. However, in many Social Web environments, the overall rating score can be decomposed into a number of "likes" and "dislikes" (with overall rating score = #likes - #dislikes). This can reveal additional information about the community perception. For instance, consider a comment receiving 10 "likes" and 0 "dislikes" versus a comment receiving 100 "likes" and 90 "dislikes". Although the overall rating is the same for both comments (i.e. +10), the content of the latter comment is likely to be more controversial. More generally, in this section we study *controversial comments* which attract a more balanced number of positive and negative votes versus rather *non-controversial comments* where either the positive or the negative votes are dominant. Table 3.10 shows some hand-picked comments from our Yahoo! News set to illustrate both classes of comments. For instance, those comments that are supporting/criticizing either one of the democrat or republican leaders in the US are equally liked and disliked. In contrast, a general criticism about politicians is liked by almost everybody, and praising Bin Laden is disliked by almost all of the raters.

In this section, we first provide an analysis of controversial comments covering various aspects, and then focus on developing models for automatically detecting controversial comments. Retrieving such comments can be especially interesting for opinion researchers and journalists as it allows them to identify, in advance, comments and topics that divide the community. Note that our discussion is restricted to Yahoo! News comments because YouTube did not provide separate numbers for "likes" and "dislikes" at the time of crawling.

3.5.1 Analysis of "likes" and "dislikes"

We first want to study how the different proportions of "likes" and "dislikes" are distributed in Yahoo! News. For a comment c containing at least one rating, we define the *comment approval ratio* (Φ) as $\Phi(c) = \frac{l_c}{l_c+d_c}$, where l_c (d_c) represents the number of likes (dislikes) for comment c. A ratio close to 0 means that the comment is totally rejected by the community and 1 indicates complete approval/acceptance. In contrast,

Table 3.10: Examples of comments belonging to the categories *"controversial"* and *"non-controversial"*.

Likes	Dislikes	Text
		Controversial Comments
24	16	`Why do the republicants hate working class America so much? OBAMA 2012`
32	32	`Bush....gasoline $1.87/gal Obama...gasoline $3.47/gal`
16	20	`The Tea Party hates two things. 1. Being called racist. 2. Black people`
10	15	`For some reason, a lot of you thing that rich people pay NO taxes? They pay taxes even though 50% of Americans do not. What Obama wants to do is RAISE their taxes. That's not fair. Let's make sure everyone pays taxes and politicians use tax money in a sensible way before we raise taxes on a few.`
22	27	`Sounds a whole lot like how scientists are treated who don't believe man-made global warming alarmists.`
		Non-controversial Comments
118	2	`we have the best politicians.......THAT MONEY CAN BUY!!`
21	0	`Politicians should be required to wear the logos of their corporate sponsors like race car drivers do... there would be a lot less confusion about who they're actually representing.`
34	1	`Washington DC has over 41000 lobbyist sorry guys business as usual`
104	9	`Occupy Washington DC !`
2	32	`Osama bin Ladin will forever be remembered as the man who brought America to it's knees. 9/11 was a blessing`
117	13	`One day I hope everyone who lost someone in this disaster can be at rest. Im so sorry for the ones who still hurt and ask GOD to put u at rest just knowing they r with him now and SAFE.GOD BLESS AMERICIA.`

$\Phi(c)$ values around 0.5 correspond to *controversial comments* that received a balanced number of "likes" and "dislikes". Since comments with a higher number of ratings can provide stronger evidence for the users' opinions in comparison to those with only few ratings, we define a threshold θ for the total number of likes and dislikes received by a comment. For our study we chose θ= 15, 20, and 25.

Figure 3.15a shows the distribution of Yahoo! News comments with respect to their Φ values. We observe that the number of comments increases with higher values

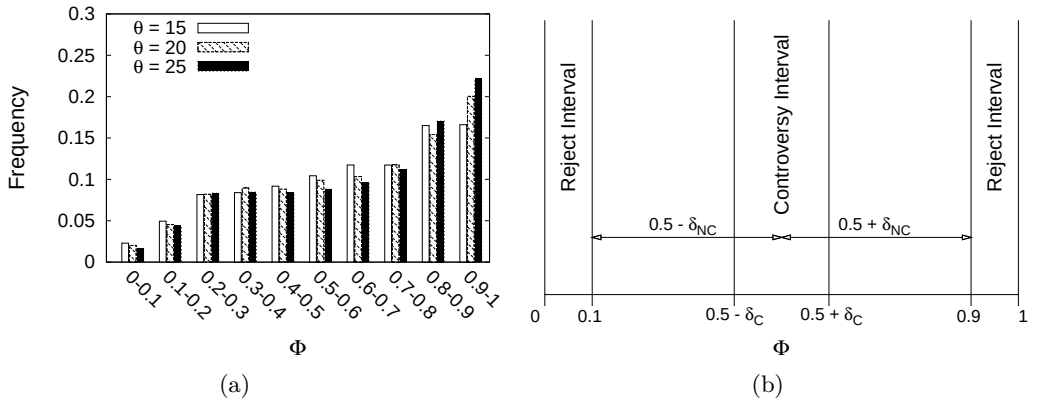


Figure 3.15: (a) Distribution of number of comments per comment approval (Φ) intervals for distinct thresholds θ for the number of received ratings. (b) Controversy interval vs. accepted (positive) and not accepted (negative) intervals.

of Φ, which is expected given our earlier findings that comment ratings are skewed towards positivity. In particular, while highly-disliked comments (the first bin in the plot) constitute less than 5%, highly-liked comments (the last bin) constitute more than 15% of all comments in a given set. The positivity in ratings becomes even more dominant if we focus on comments with a larger number of ratings (i.e. higher values of θ). Regardless of the θ threshold, comments with $\Phi \in [0.4, 0.6]$ (i.e. around 0.5) add up to 20% of all comments. This shows that the number of comments causing controversy among users is relatively large.

As depicted in Figure 3.15b, based on the comment approval ratio, Φ, we more formally define a comment c as controversial if $0.5 - \delta_C \leq \Phi(c) \leq 0.5 + \delta_C$ where $\delta_C \in [0, 0.5]$ defines the boundaries of the *controversy interval*. Analogously, we define non-controversial comments as comments c with $0.5 - \delta_{NC} \leq \Phi(c) \leq 0.5 + \delta_{NC}$ where δ_{NC} specifies the required distance of a comment's Φ value from 0.5. In the following, unless stated otherwise, we set δ_C equal to 0.1, i.e., we consider comments for which Φ values fall into the range $[0.4, 0.6]$ as controversial comments. For the non-controversial comments, we study δ_{NC} values of 0.2, 0.3, and 0.4, with larger values of δ_{NC} corresponding to more restrictive thresholds for considering comments as non-controversial.

Figure 3.16 shows the distribution of controversial comments across the news stories for all stories with at least 1 controversial comment (absolute values, $\delta_C = 0.1$). Overall, we observe a considerable amount of comments matching our definition of controversial comments. For instance, for $\theta = 15$, around 15% percent of all stories contain at least one controversial comment posted for them. As expected the controversial comments follow a Zipf-like distribution.

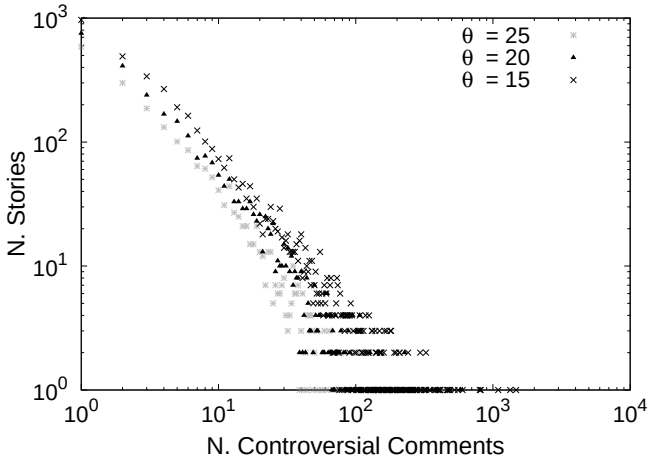


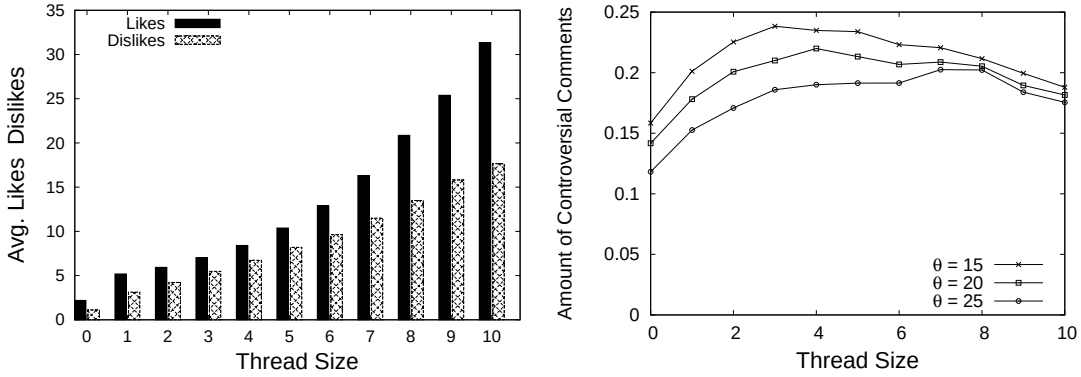
Figure 3.16: Distribution of controversial comments for distinct thresholds θ for the number of received comment ratings.

3.5.2 Term Analysis of Controversial Comments

In order to examine the differences in language and vocabulary usage between controversial and non-controversial comments we conducted a discriminative term analysis. We set δ_C to 0.1, δ_{NC} to 0.4 and θ to 25, and, using Mutual Information, we computed a ranked list of stemmed terms for approx. 6,000 comments from each of the two classes. Table 3.11 shows the top-20 stemmed terms extracted for each set. Many of the controversial comments contain terms related to political parties/entities involved in US presidential elections (*obama*, *republican*, *democrat*, *bush*) or terms expressing strong emotions (*believe*, *hate*). We conducted a manual inspection of comments and found that the latter type of terms is often used in conjunction with political entities, as there exist several bigrams such as *blame obama*, *vote obama*, and *hate bush*. Non-controversial comments, on the other hand, also contain terms related to politics; however, those are rather general terms such as *washington*, *politician*, and *govern* that are not specific to any political group. Note that, by definition, the set of non-controversial comments are those found at the two extremes of the spectrum defined by our Φ values. This explains why the term list for the non-controversial comments include words like *corrupt* and *hope*, which might be extracted from the comments that are either rated "negative" or "positive" by a vast majority. Another interesting example in the non-controversial term list is the word *bank*; our manual inspection of corresponding comments revealed that banks are often criticized because of their role in the financial crisis, and these comments are approved by a large majority of the users.

Table 3.11: Top-20 terms according to their MI values for controversial vs. non-controversial comments.

Terms for Controversial Comments		**Terms for Non-Controversial Comments**	
obama	muslim	politician	need
republican	want	govern	month
liber	black	congress	law
bush	america	polit	mother
presid	right	time	help
tea	rich	money	food
parti	blame	bank	limit
gop	hate	washington	famili
2012	fact	hope	buy
democrat	believ	corrupt	day



(a) Average Likes and Dislikes (b) Fraction of Controversial Comments

Figure 3.17: Comment ratings and controversy with respect to thread size.

3.5.3 Analysis of Ratings for Comment Threads

In Section 3.4, we showed that comments receiving a larger number of replies have more positive ratings on the average. In this section we extend that analysis by investigating the *controversy* of comments resulting in discussion threads.

Figure 3.17a shows the average number of likes and dislikes of seed comments for different discussion thread sizes. We observe that, on average, seed comments for longer threads receive a larger number of likes and dislikes. Additionally, the number of likes grows faster than the number of dislikes, i.e., the comments that triggered longer discussions tend to be associated with more positive ratings. We also notice that the gap between likes and dislikes increases for larger thread sizes. Figure 3.17b shows the fraction of controversial seed comments (with Φ values in $[0.4, 0.6]$) with respect to the size of the initiated threads. The initial increase in the amount of controversial comments is expected, and shows that comments initiating discussions tend to be more

Table 3.12: BEPs for controversial comment prediction.

θ	δ_{NC}=0.2			δ_{NC}=0.3			δ_{NC}=0.4		
	BEP	#stories	#comments	BEP	#stories	#comments	BEP	#stories	#comments
15	0.571	3,690	13,6000	0.59	3,439	101,000	0.649	2,861	52,000
20	0.579	2,514	57,000	0.623	2,272	42,000	0.668	1,891	24,000
25	0.589	1,752	28,000	0.633	1,564	21,000	0.679	1,258	12,000

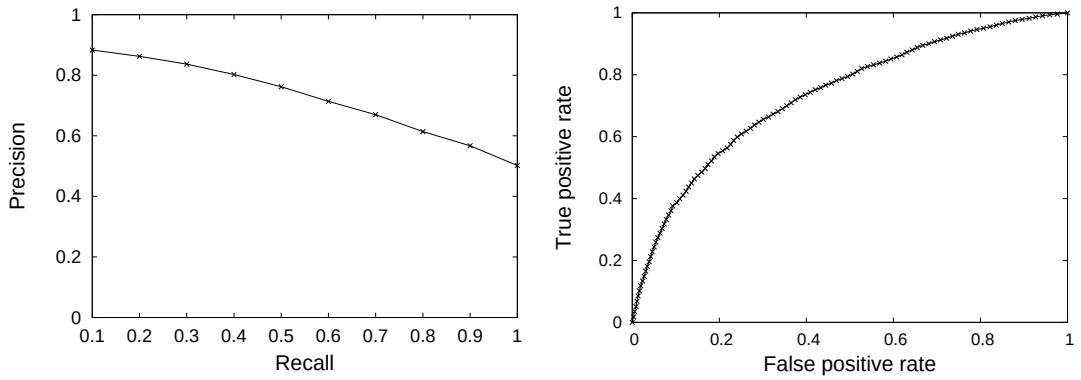
controversial. Interestingly, the fraction of controversial comments reaches a maximum between thread size 3 and 5, and slowly decreases afterwards.

3.5.4 Predicting Controversial Comments

Can we predict controversial comments, i.e. comments that receive a comparable number of likes and dislikes from the community at the same time? In order to explore this, we built binary SVM classifiers based on the textual content of the comments. More specifically, from each story, we first extracted an equal number of n controversial and non-controversial comments, with n being the smaller of cardinalities of the controversial/non-controversial comments, and controversy determined by the definitions provided in Section 3.5.1. The overall set of controversial and non-controversial comments formed the positive and negative instances for the learning algorithm. We fixed the controversy interval in our experiments to be [0.4,0.6] (corresponding to $\delta_C = 0.1$), and varied δ_{NC} and θ to explore the impact of a wider range of Φ values for the negative class and the total number of ratings, respectively. We constructed feature vectors for comments as described in Section 3.3.5. We tested the classifier performance using 5-fold cross-validation with dataset sizes shown in Table 3.12 (keeping 80% of the data for training and 20% for testing in each of the 5 runs) and using LIBSVM with cost parameter C=0.1. We repeated this experiment for different values of the parameters δ_{NC} and θ.

The results for the precision-recall break-even-points (BEPs) are shown in Table 3.12 along with the total number of stories and comments used for each configuration. We can observe two main trends: First, for a given θ, increasing δ_{NC} improves the classification performance. This is reasonable, as larger δ_{NC} values restrict non-controversial comments to a narrower band where the ratio of likes and dislikes differs substantially; as a consequence, the classifier can distinguish these comments from the controversial ones more easily. Secondly, as expected, comments with a higher number of ratings (corresponding to larger values of threshold θ) provide stronger evidence while learning to classify controversy.

The ROC curve is shown in Figure 3.18b. We obtain a value of 0.733 for the



(a) Precision-recall curve (b) ROC Curve

Figure 3.18: Precision-recall and ROC curve for the classification of controversial comments (δ_{NC}=0.4).

AUC (Area Under the ROC Curve) and an accuracy of 0.649. Figure 3.18a shows the detailed precision-recall curve for $\theta = 25$ and $\delta_{NC} = 0.4$. While the BEP value is relatively low, trading recall for precision leads to applicable results. For instance, for the non-controversial set corresponding to comments with Φ values in the first and last 10% bands ($\delta_{NC} = 0.4$), the precision is 0.859 for a recall level of 0.1 and greater than 0.8 up to a recall level of 0.3. This is useful for application scenarios such as displaying a relatively small number of potentially controversial comments at visible ranks.

3.6 Users Commenting on Social Web Environments

In this section, we will focus on the *users* commenting on content in social web environments. Commenting tools in social websites are mostly used to share legitimate opinions and feelings. Nevertheless, it is also common to find users that abuse this mechanism in various ways. These include posting links to external web pages aiming at increasing their visibility (i.e. spamming), or "posting disruptive, false or offensive comments to fool and provoke other users" (i.e. trolling [79]). We conduct an exploratory analysis of the presence of troll users (*trolls*) in social websites, and study methods for automatically detecting potential trolls based on the textual content of their comment history.

3.6.1 Finding Trolls

Our main datasource for this analysis is the YouTube collection. Yahoo! News allows participants to use non-unique identifiers for commenting, which renders the task of

modeling troll behavior unfeasible. Given this limitation of Yahoo! News, we decided to collect an additional dataset in order to provide a more comprehensive analysis of trolling characteristics in social websites. In particular, we crawled the popular technology news website Slashdot[7], as for this site it is possible to easily obtain a set of manually classified trolls and their comments. To this end, we followed the procedure proposed by [79] and extracted a set of troll users from a special user account, called *No More Trolls*, which tags all *known* trolls as its *foes* to help other users avoid them. We crawled the 24 most recent comments (i.e. the maximum number of comments per user shown by Slashdot) from all users listed as trolls. The resulting collection includes 200 users and 4310 comments. An additional random sample of 200 users not contained in the *No More Trolls* list was crawled to represent the negative class, i.e. "non-trolls".

Extracting a comparable number of trolls from the YouTube dataset is not straightforward. First, troll detection requires manually assessing the content of each user's comments as YouTube does not provide a list of troll users flagged by the community. Second, the proportion of trolls is significantly lower than that of legitimate users [79]. Therefore, identifying a comparable amount of trolls in YouTube using a random sampling strategy would require manually inspecting comments from thousands of users. To decrease the manual effort required, we used a simple heuristic to increase the chance of finding trolls in our sample by means of the *user approval ratio* $\Psi := \frac{pos(u)}{pos(u)+neg(u)}$, where $pos(u)$ and $neg(u)$ denote the number of positively and negatively rated comments for a given user u, respectively. Low values of this ratio indicate strong rejection by the community for the comments of a particular user, while high values indicate general acceptance of the user's opinions. We used this metric to sample YouTube users by randomly selecting 500 users with $\Psi(u) \in [0, 0.1]$ under the assumption that a significant number of trolls would fall into this interval. In order to obtain a set containing more non-troll users we also sampled a set of 500 users with approval ratio $\Psi(u) \in [0.1, 1]$. The final set of 1,000 users was then *manually* annotated by the second author of our published paper [121] using the following three labels based on the content of their comments: "troll", "non-troll", or "unknown".

3.6.2 Trolls and Community Ratings

Can comment ratings serve as an indicator for trolling behavior? Figure 3.19 shows the distribution of troll and non-troll users in YouTube with respect to the user approval ratios Ψ. This figure clearly illustrates the higher proportion of trolls found in the $[0, 0.1]$ Ψ range, as compared with the proportion at higher levels of Ψ. This result provides empirical support for the heuristic chosen in our sampling strategy. We also

[7] http://www.slashdot.org/

observe a large percentage of trolls in the $[0.1, 0.2]$ range, whereas just a tiny fraction of users are found to be trolls for $\Psi > 0.2$. This confirms the intuition that comment ratings serve as good proxies for troll identification in online communities.

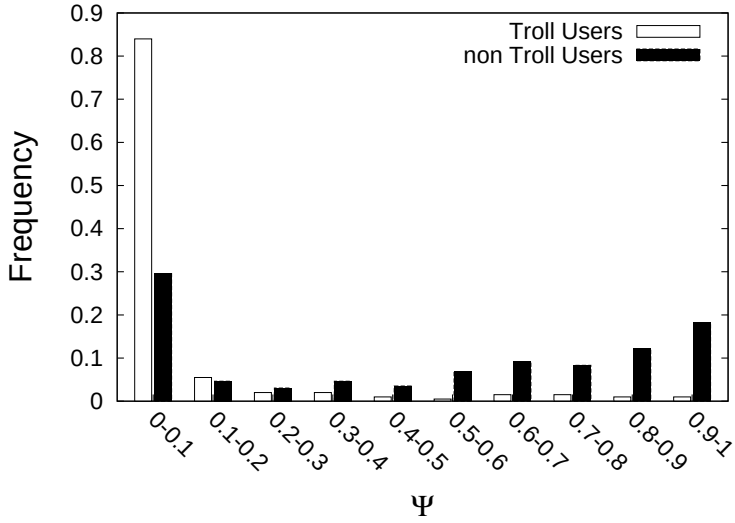


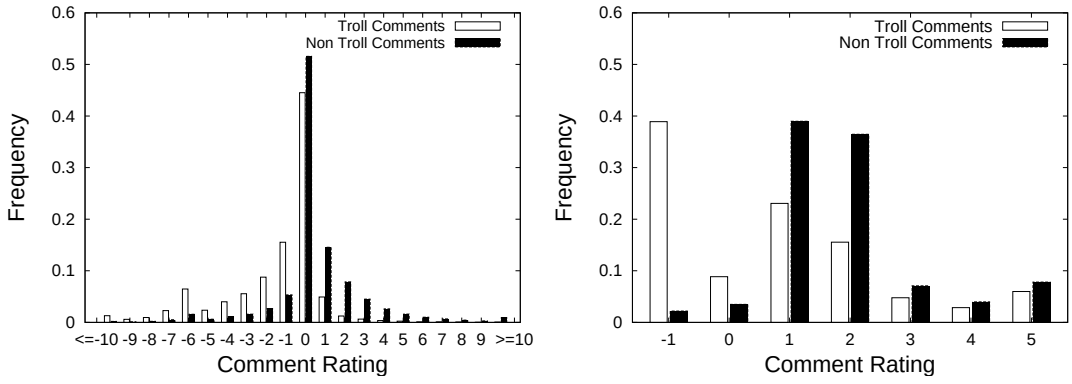
Figure 3.19: Distribution of troll and non-troll users in YouTube with respect to user approval ratio (Ψ) intervals.

Figure 3.20 shows the distribution of comment ratings from YouTube (Figure 3.20a) and Slashdot (Figure 3.20b). Note that our sampling strategy for detecting trolls in YouTube is biased towards low rated comments, as 50% of the comments were chosen from Ψ values in the range $[0, 0.1]$. As illustrated in Figure 3.19, this bias significantly affects the distribution of non-troll comment ratings, but has just a marginal effect on the distribution of troll comment ratings as they mostly feature low rating values. Therefore, we address our sampling bias by comparing the ratings of comments from trolls in this sample with ratings from comments in the whole dataset (including troll and non-troll users). Both plots show a clear trend of comments from troll users having lower ratings than comments from non-troll users in both communities.

3.6.3 Content-based Troll Prediction

How does vocabulary usage differ for troll and non-troll users, and can the textual content of comments be leveraged for detecting trolls?

We compared the most discriminative terms of the comments from troll and non-troll users in YouTube and Slashdot. For each dataset, we randomly selected 200 troll and 200 non-troll users and extracted 24 comments sampled uniformly at random. Analogously to the term analyses in previous sections we computed the top-ranked (stemmed) terms with respect to the Mutual Information measure. Table 3.13 reveals



(a) YouTube (b) Slashdot

Figure 3.20: Comment rating distribution for comments from troll users and non-troll users in (a) YouTube and, (b) Slashdot.

Table 3.13: Top-20 terms according to their MI values for troll vs. non-troll comments.

Terms for Troll Comments				**Terms for Non-Troll Comments**			
YouTube		Slashdot		YouTube		Slashdot	
fuck	dick	fuck	bush	love	happen	use	pretti
shit	stupid	post	vomit	look	use	think	peopl
suck	young	troll	failur	good	thought	work	time
ass	hey	slashdot	nigger	miss	great	http	agre
white	cunt	linux	enjoy	time	doe	year	game
nigger	black	shit	ass	awesom	thank	thing	look
bitch	retard	fail	love	think	clay	problem	actual
free	cock	die	cybernet	agre	hot	compani	phone
gay	watch	gay	crapflood	lol	end	doe	realli
u	jew	fp	clit	s	govern	know	probabl

that the terms used in comments from trolls are very similar for YouTube and Slashdot, and these terms are mostly offensive. Despite exhibiting less similarity, the term lists for non-troll users generally include more positive terms such as *good*, *love*, *beauty* (in YouTube) and *like*, *pretti*, *agre* (in Slashdot). Some of the Slashdot terms look counterintuitive at first sight. For instance, terms used by trolls in Slashdot include *linux*, *slashdot*, or *cybernet*. Further inspection of the data revealed that trolls often use them as their target for complains and insults (e.g. *"linux is a failure"*). On the other hand, we notice that, other than in YouTube, terms in the non-troll category seem to be slightly less positive (or even neutral) in Slashdot (e.g. *http*, *game*, *phone*). This could be related to the fact that comments in Slashdot are mostly used to engage in technical discussions. An interesting observation is that *http* is a discriminative term for non-troll comments in Slashdot; this mainly corresponds to posting links in comments which are often appreciated by the community.

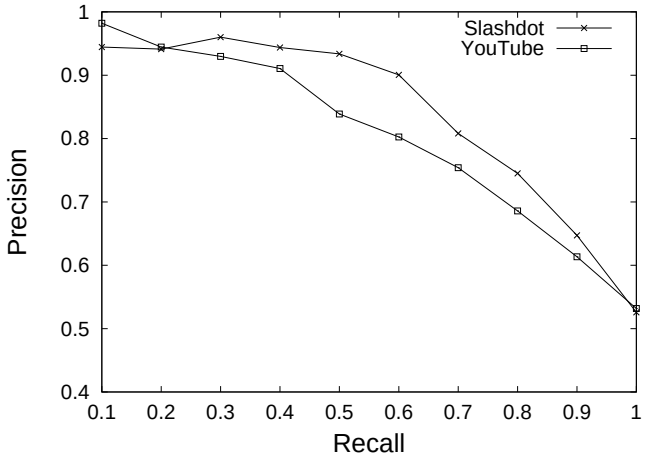


Figure 3.21: Precision-recall curves for troll detection in the YouTube and Slashdot datasets.

The difference in the terms chosen by trolls and normal users within their comments encouraged us to study the possibility of training SVM classifiers (using LIBSVM with linear kernel and cost parameter C=0.1) to predict trolls by using the textual content of their comments. Comments from one user were merged into a single "virtual" comment; feature vectors were then constructed as described in Section 3.3.5. We used 200 users (and 4,800 comments) for each class of users, and a 50-50 split with 2-fold cross validation to report the average classification performance. We used 2-fold CV due to the relatively small size of the datasets. Splitting the set into a larger number of folders would have rendered the computation of the individual precision-recall curves in each CV run very unstable (especially due to the very small number of documents that would be involved for computing the low-recall part of the curves).

Figure 3.21 shows the precision-recall curves for predicting troll users from YouTube and Slashdot. We observe BEP values of 0.682 and 0.742 for YouTube and Slashdot, respectively. Our findings reveal that the precision is greater than 0.8 up to a recall value of 50% for both datasets. The relatively large difference in the classification performance for the datasets suggests inherent differences in the communities and their commenting behavior, as previously observed in this chapter. The YouTube collection mainly contains short and unelaborate opinions that provide fewer cues for the correct classification of users as trolls.

Note that troll detection applications should be tuned to seek high precision. Automatic troll detection needs to avoid censoring legitimate users, as this could result in user frustration and, ultimately, community destruction. We believe that troll detection should be carefully assessed by human supervisors to avoid any possibility of user loss, as users are the main asset of social websites.

3.7 Summary and Contributions

Our key contributions in this chapter are as follows.

We provided an in-depth analysis of comment-centric feedback in two prominent social websites, YouTube and Yahoo! News, aiming at achieving a better understanding of comment-centric community feedback on the Social Web. For troll detection we conducted additional experiments on Slashdot as this dataset has been explored in this context in previous works.

A couple of results derive from the presented studies. Community meta-feedback provided through comment ratings is indeed dependent on characteristics of the comments content such as orientation of opinions; we observed that positive opinions expressed in the comments attract positive community feedback, and vice versa. Comment content helps predicting various types of community feedback, such as overall comment rating, ratio of likes and dislikes for the comments, likelihood of comments triggering replies, and further participation from the community. These results are clearly more prominent in the YouTube community, where the abuse of language occurs significantly more often as compared to Yahoo! News, partly because of stricter comment filtering policies in the Yahoo! News system. We also showed that comment-centric community feedback can help identifying polarizing content, that is, content that generates rich discussions between community members with contrary opinions.

Finally, we studied the characteristics of users, specifically trolls, commenting on content in social web environments. We found that comment content can be leveraged to effectively identify troll users.

4 Social Feedback

In addition to commenting, rating and replying comments, Web 2.0 platforms such as YouTube provide additional means for the users to interact with the content (e.g., via likes, dislikes, favorites). This results in a vast amount of social feedback available for the multimedia content shared through the Web 2.0 platforms. However, the potential of such social features associated with shared content still remains unexplored in the context of information retrieval. In this chapter, we first study the social features that are associated with the top-ranked videos retrieved from the YouTube video sharing site for the real user queries. Our analysis considers both raw and derived social features. Next, we investigate the effectiveness of each such feature for video retrieval and the correlation between the features. Finally, we investigate the impact of the social features on the video retrieval effectiveness using state-of-the-art learning to rank approaches. In order to identify the most effective features, we adopt a new feature selection strategy based on the Maximal Marginal Relevance (MMR) method, as well as utilizing an existing strategy.

4.1 Related Work

Among the social features associated with the shared content, the lion's share of research interest is devoted to the user comments due to their potential to improve the performance in several scenarios. In a recent survey, Potthast et al. categorize the comment related tasks as comment-targeting and comment-exploiting [108]. The works that aim to rank [75] or diversify the comments [61] and predict their ratings [120] fall into the former group. In the latter category, there is a large body of works that utilize the comments for various purposes, such as summarizing the blog posts [76], classification of YouTube videos [56], predicting the content popularity [96, 131, 145] and recommending the related content items [118].

Despite the large number of works focusing on the comments, only a few of them

investigate their potential to improve the retrieval effectiveness, and usually in isolation, i.e., independently from the other social and basic features. In one of the earliest studies, Mishne and Glance investigate the impact of comments on the retrieval performance for weblogs and report that employing comments does not improve the precision, but helps to retrieve both relevant and highly discussed blog posts [96]. The user comments in MySpace are exploited for ranking the artists [65]. In [115], comments are leveraged for the aesthetic-aware re-ranking of image search results. The closest work to ours is [146], which utilizes YouTube comments for the video retrieval. However, their work is only limited to the experimenting the comment feature within the known-item retrieval scenario. To the best of our knowledge, we are the first to investigate the retrieval effectiveness of a rich set of social features in combination with the basic ones within a realistic search scenario.

In [88], the authors provide an exhaustive survey for a variety of LETOR approaches which appear in the literature. The approaches can be categorized into three categories, namely, point-wise, pair-wise and list-wise depending on their loss function. In this thesis, we employ state-of-the-art representatives from each category. We present a unique dataset that includes a total of 100 popular and tail queries submitted to YouTube and around 10,000 relevance annotations for the results of these queries. Furthermore, different from all these commercial and academic datasets, we define various social features obtained from the real YouTube results in addition to the typical basic features. To the best of our knowledge, no earlier study investigates the retrieval effectiveness of a set of social and basic features in combination within a LETOR framework.

4.2 Data Gathering, Methods and Characteristics

The first challenge in investigating the impact of social features in ranking YouTube videos is creating a dataset based on *real* user queries. Previous studies typically obtain samples of YouTube content by running crawlers that are seeded with some generic queries (e.g., the queries from Google's Zeitgeist archive [120] or terms from the blogs and RSS fields [128]). Different from these works, we employ a methodology for creating two different query sets including the popular and rare queries that are actually submitted by YouTube users. In what follows, we describe our query sets and the top-ranked videos retrieved for these queries.

4.2.1 Query Sets

In order to construct a representative sample of real user queries, we made use of the auto-completion based suggestion service specialized for the YouTube domain from a

major search engine. These instant suggestions are typically based on the previous queries submitted by other users [122, 11]. We submitted all possible combinations of two-letter prefixes in English (i.e., aa, ab, ..., zz) and collected the top-10 query suggestions for each such prefix (e.g., "**aa**liyah", "**aa**ron carter", "**ab**ba dancing queen", etc.) in a similar fashion to [29]. This process yielded a set of 7,000 suggestions, from which a subset of 1,447 queries is sampled uniformly at random (to avoid overloading YouTube servers with an excessive number of requests in the next steps). We call this latter set as *popular* queries (denoted as Q_P) because these suggestions, appearing after typing any two letters, are very likely to be submitted a large number of times in the past (we further justify this claim through a volume analysis later in this section).

Previous research on web search logs shows that the query frequency distribution follows the power law and hence, a large fraction of the queries in the long-tail are rare [122]. As the commercial search engines are all good in answering the highly popular queries, the competition in the search market is becoming more focused on queries in the long tail of the distribution (e.g., see [148]). Therefore, for an exhaustive analysis of the impact of social features in ranking, we also created another set of queries that include such infrequent queries, referred to as *tail* queries hereafter. To this end, we repeated the above procedure for a second round, but this time for each query in Q_P, we submitted the complete query string followed by a combination of any two letters. For instance, for the popular query "csi miami", this procedure returns the suggestions "csi miami **th**e best defense ian somerhalder", "csi miami **to** kill a predator ending", and "csi miami **ri**ck stetler arrested" for the prefixes **th**, **to** and **ri**, respectively. Since our goal is creating a set of tail queries, for each query in Q_P, we chose the longest suggestion (in terms of the number of words), as previous works show that the tail queries are typically longer than the popular queries. This process yielded a set of 1,336 queries (as for some queries in Q_P there were no additional suggestions) that is denoted as Q_T.

Intuitively, the queries in Q_P serve as a representative sample for the head and torso queries submitted to YouTube, whereas those in Q_T represent the tail queries. We further justify this intuition by conducting a query volume analysis using two well-known tools, namely, Google Trends[1] and Google Keyword Tool[2]. In particular, we first conduct a preliminary analysis with the Google Trends Tool for a subset of our queries, and being encouraged by the results, we then employ the Google Keyword Tool to obtain the search volume for all of the queries in our query sets. While this is less accurate than using the actual YouTube query logs, given that such logs are never publicly disclosed, the volume information from the largest search engine of the world

[1] http://www.google.com/trends/?
[2] https://adwords.google.com/o/KeywordTool

should serve as a fairly good approximation.

4.2.2 Query Volume Analysis

Query Volume Analysis with Google Trends Google Trends is a tool that enables the users to explore the traffic patterns of queries over time and geography. It does not provide actual query frequencies, but only reports a relative volume distribution over time, which is normalized with respect to the peak volume observed on a particular date. This tool also allows limiting the scope of trend analysis to either "Web search" or "YouTube search", a feature that we exploit in the following analysis.

We sampled uniformly at random 250 queries from each of the popular and tail query sets, and for each query, we obtained the volume information from Google Trends by limiting the scope first to "web search" and then to "YouTube search". During this process, we set the page language to English, log off from all Google products to avoid any personalization effect and set the region parameter to "worldwide". Note that, this is a manual procedure as Google discourages all sorts of automatic accesses, and that is why we restricted our analysis to around 18% of the total number of queries (i.e., 2,810) in our query sets. We summarize our findings as follows:

- *Analysis of the trends for YouTube search:* We observe that for 81.6% of the tail queries, the tool displays the message "Not enough search volume to show graphs". In other words, only 18.4% of the tail queries have an adequate search volume in YouTube that worths to report. In contrast, 98% of the popular queries are frequent enough to yield a trend plot. This is a positive finding implying that our popular and tail query sets essentially include queries from their respective classes.

- *Comparison of the trends for the web search and YouTube search:* Secondly, we investigate whether the web search trends and YouTube search trends are correlated for our queries. We observe that 78.8% of the tail queries are reported to have *not enough* search volume in *both* web and YouTube domains. In contrast, 87.2% of the popular queries yield a volume plot for *both* web search and YouTube search. Furthermore, we observed that the trend plots obtained for the web search and YouTube are quite similar for the latter queries. This analysis implies that the search trends are similar in the web and YouTube search for our queries. Encouraged by this finding, in the next analysis, we employ the Google Keyword tool to obtain the actual volume of our queries in the web search, as they may serve as a fairly good approximation of the volume in YouTube.

Table 4.1: Query volume characteristics for the popular and tail queries.

	Popular	Tail
Average	5,823,495.10	291.02
First Quartile	1,900	0
Median	33,100	0
Third Quartile	450,000	5

Query Volume Analysis with the Google Keyword Tool The Google Keyword Tool is developed to assist choosing the appropriate ad words and it provides the local and global monthly average search volume (over the last 12 months) of a query for the selected countries, languages and devices (i.e., desktop and laptop devices, mobile devices, etc.). As there is no API to access this service, which is an understandable policy aiming to discourage spammers and black hat marketers, we opted for a crowdsourcing solution. For each query in $Q_P \cup Q_T$, we created a Human Intelligence Task (HIT) that asks workers in Amazon Mechanical Turk (AMT)[3] to submit the given query to the Google Keyword Tool and to enter the returned volume to the corresponding field in the HIT. By this way, we collected the global monthly average search volume for a total of 2,810 $(1,447+1,363)$ queries. We also sampled a subset of 50 queries from each set and repeated the same process ourselves, to verify the reliability of AMT results. It turned out that AMT results are quite reliable for this task; as among the set of 100 queries we tried, there were only a couple of queries for which the volume information reported by the Turkers differ from what we got.

We noticed that a large fraction of the tail queries have no volume information, which implies that they are extremely rare, and even the remaining ones have very low frequencies in comparison to those in the popular set. In Table 4.1 we show the statistics for the queries in our Q_P and Q_T sets. These values justify our intuitive methodology for constructing our query sets as the queries in Q_P are found to be three order of magnitudes more popular than those in Q_T. In other words, we can safely claim that Q_P and Q_T are representative samples of, respectively, popular (i.e., head and torso) and tail queries that are submitted to YouTube. We provide some illustrative examples from both query sets in Table 4.2.

Query Results For each q in Q_P, we obtained the top-100 result videos (denoted as R_q) from YouTube API along with the available metadata fields (see Table 4.3) in late 2011. This process resulted in a superset of 138K videos, i.e., around 95 videos per query are retrieved. Among these videos, 132,697 of them are unique (i.e., only 4.3% of all videos overlap among different query results). The set of unique videos

[3] https://www.mturk.com

Table 4.2: Example popular and tail queries with the global monthly average search volume.

Popular		Tail	
Query	Volume	Query	Volume
adele	13,600,000	adele turning tables live at the royal albert hall lyrics	12
kfc chicken	5,000,000	how to make kfc chicken burger at home	16
coldplay paradise	550,000	coldplay paradise remix fedde le grand radio edit	170

Table 4.3: Metadata fields stored for each video.

Metadata	Notation	Metadata	Notation
No. of views	$W(v)$	Title	$TitleText(v)$
No. of likes	$L(v)$	Tags	$TagText(v)$
No. of dislikes	$D(v)$	Description	$DescText(v)$
No. of comments	$C(v)$	Comments	$CommentText(v)$
Uploader	$U(v)$	Age	$G(v)$

for the popular queries is denoted as V_P. For the tail queries, we repeated the same procedure (in late 2012). The resulting set of unique videos for the tail queries, V_T, includes 63,693 videos.

In addition to the metadata fields directly available via the API, we crawled up to 10,000 most recent comments that are posted for each video from the actual HTML responses of YouTube. In this way, we obtained around 33 million comments posted for around 86K unique videos in our dataset. This is a fairly large set of comments as the recent works also employ similar (e.g., up to 1,000 comments for 40K videos in [128]) or smaller number of comments (e.g., a total of 6.1 million comments in [120]).

Finally, we also constructed the profiles of the users who uploaded the videos in V_P and V_T. To this end, for each user u, we again crawled the HTML pages to obtain the number of uploaded videos, number of subscribers (i.e., the number of users who are following the user u), and total number of views for the content uploaded by the user u. We ended up with the profiles for 85,068 and 44,646 unique users, denoted as U_P and U_T, for the videos in V_P and V_T, respectively.

Note that, the metadata fields in Table 4.3 (other than $TitleText$, $TagText$ and $DescText$ that are related to the basic features) constitute the raw social features, from which we derive various social features as described in detail in Section 4.3.

4.2.3 Statistics for the Result Videos

In Table 4.4, we provide the basic statistics on the numeric metadata fields computed over the videos in V_P and V_T. We find that most of the statistical figures (i.e., the averages and standard deviations) computed over the popular video set do not differ remarkably from those computed for the tail video set. However, for the popular queries, the video with the maximum number of views is watched 394,184,702 times, almost doubling the view count of the most-viewed video for the tail queries, i.e., 205,964,304. A similar trend is also observed for the maximum number of comments and dislikes, but not likes. These findings imply that from a set-based perspective, the top-100 videos retrieved for the popular and tail queries exhibit similar characteristics. However, the situation changes when we take the video rank into account for a finer grain analysis (especially for the top-10 videos), as we discuss in Section 4.2.4.

From Table 4.4, we further observe that some of the popularity-related metadata statistics seem to be much higher than those obtained for the YouTube crawls reported in the literature. For instance, the average view counts are greater than 400,000 for both of our video sets, which is an order of magnitude larger than the figures reported in some recent works (e.g., see Table 1 in [135] and Table 2 in [36]). This might be due to the differences in the data collection procedure; we employ a set of real user queries and get their top-ranked results while other works obtain a sample by crawling the user profiles along with the uploaded videos [135] and/or employing artificial queries to seed their crawls.

Table 4.4 also shows that the average number of comments in our collection is around 0.18% of the average number of views (computed separately for both of the popular and tail query sets), i.e., very close to 0.16% reported by Cha et al. in 2009 [25], but interestingly, less than 0.5% reported by another recent study [128]. As a final remark, we observe that the standard deviation values are rather high for all metadata fields

Table 4.4: Metadata statistics for the videos retrieved for the popular and tail queries. Lengths of the titles, tags and descriptions are in terms of the words (after stopword removal).

	Avg		Max		Stddev	
	Popular	Tail	Popular	Tail	Popular	Tail
No. of views	423,812.92	400,206.2	394,184,702	205,964,304	3,657,104.34	3,496,825.13
No. of likes	1,427.07	1,467.82	1,123,471	1,378,652	11,833.58	15,627.8
No. of dislikes	101.02	103.67	704,922	248,943	3,025.44	2,180.79
No. of comments	759.44	730.84	1,136,181	665,214	8,418.81	9,737.75
Age	503.21	630.71	2,332	2,573	493.43	478.83
Title length	6.3	6.94	32	39	2.92	3.14
Tags length	18.5	11.03	146	62	16.47	6.52
Description length	47.54	61.67	1,185	499	92.08	74.26

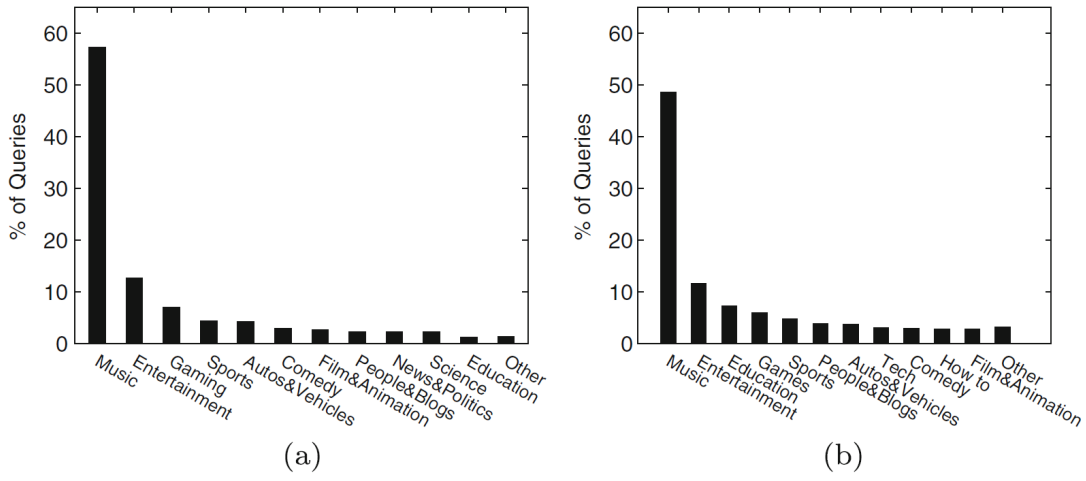
presented in Table 4.3, a result again in line with the previous findings [135].

4.2.4 Query Result Characterization using Social Features

Characteristics of YouTube queries In contrast to web search, for which publicly available query logs allow analyzing issues like the user search intentions, result distributions, etc., there exist no public query logs that can be exploited for characterizing the users interests for video search in platforms like YouTube. So, we begin with an analysis of the queries in the sets Q_P and Q_T, i.e., namely, 1,447 and 1,363 queries included in each set, respectively. Our analysis, being based on a real and representative query collection, is the first towards shedding light on the actual search interests of YouTube users.

We first classified our queries based on the YouTube provided category of the result videos. In particular, a query's category is designated as the most popular category among the videos retrieved for this query. Not surprisingly, the majority of the queries fall into the "music" category, for both popular and tail queries (Figures 4.1a and 4.1b). Figure 4.1 shows that the other popular query categories are "entertainment", "gaming" and "sports". In addition to the automatic categorization, we also conducted a manual analysis that is restricted to the popular queries (due to the heavy human labor required). We annotated the named entities appearing in the queries and found that 46% of the queries include a person entity (e.g., a singer, movie star, music band, YouTube user, etc.) and another 9% of them include a product entity.

Figures 4.2a and 4.2b provide the distribution of the number of results for our popular and tail queries, respectively. As we see, for almost half of the popular queries YouTube reports more than 10K result videos, which is rather expected due to the



(a) (b)

Figure 4.1: Category distribution of (a) popular, and (b) tail queries.

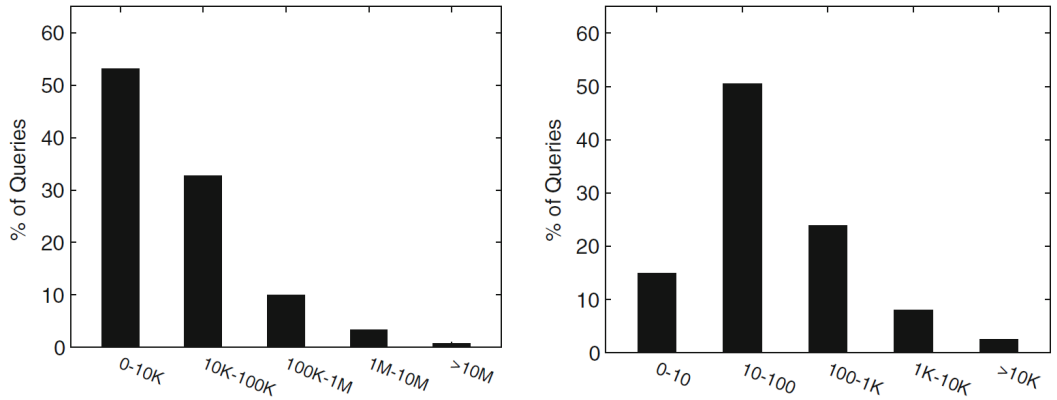


Figure 4.2: Number of results (reported by YouTube) for (a) popular, and (b) tail queries.

nature of these queries. In contrast, around 60% of the tail queries retrieve less than 100 videos. In this sense, YouTube queries exhibit a similar behavior to what has been reported in the literature for the Web search queries. For instance, Skobeltsyn et al. [123] report that the result set sizes for the tail queries (or, more specifically, cache "misses") obtained from a large Web search log are two order of magnitudes smaller than those for a general query log sample. In Figure 4.2, we present a similar finding for the popular vs. tail queries submitted to YouTube.

Characteristics of the top-ranked results In this section, we present the basic characteristics of the top-100 query results with respect to the raw social features such as the number of views, likes, dislikes and comments. While earlier studies about YouTube (e.g., [36, 25, 135]) also report statistics for some of these features, their analyses are usually over a *set* of videos (e.g., such as those we provide in Table 4.4). To the best of our knowledge, ours is the first study that provides an analysis for the social features taking into account the rank of the videos in the query results presented by YouTube.

Figure 4.3a shows the average number of views for the videos that are ranked at the i-th position in the query results, where $1 \leq i \leq 100$, for the popular and tail query sets. In general, the number of views is quite high (around 200,000 views even at rank 100), for both query sets. However, for the results of the popular queries, we see that the number of views for the top-ranked video is considerably higher than those for the others and indeed each of the the videos in the top-7 results is viewed more than 1 million times on the average. The trend is similar but less strong for the top-ranked results of the tail queries, i.e., only those videos ranked in the top-4 places have considerably larger number of views than the rest of the results.

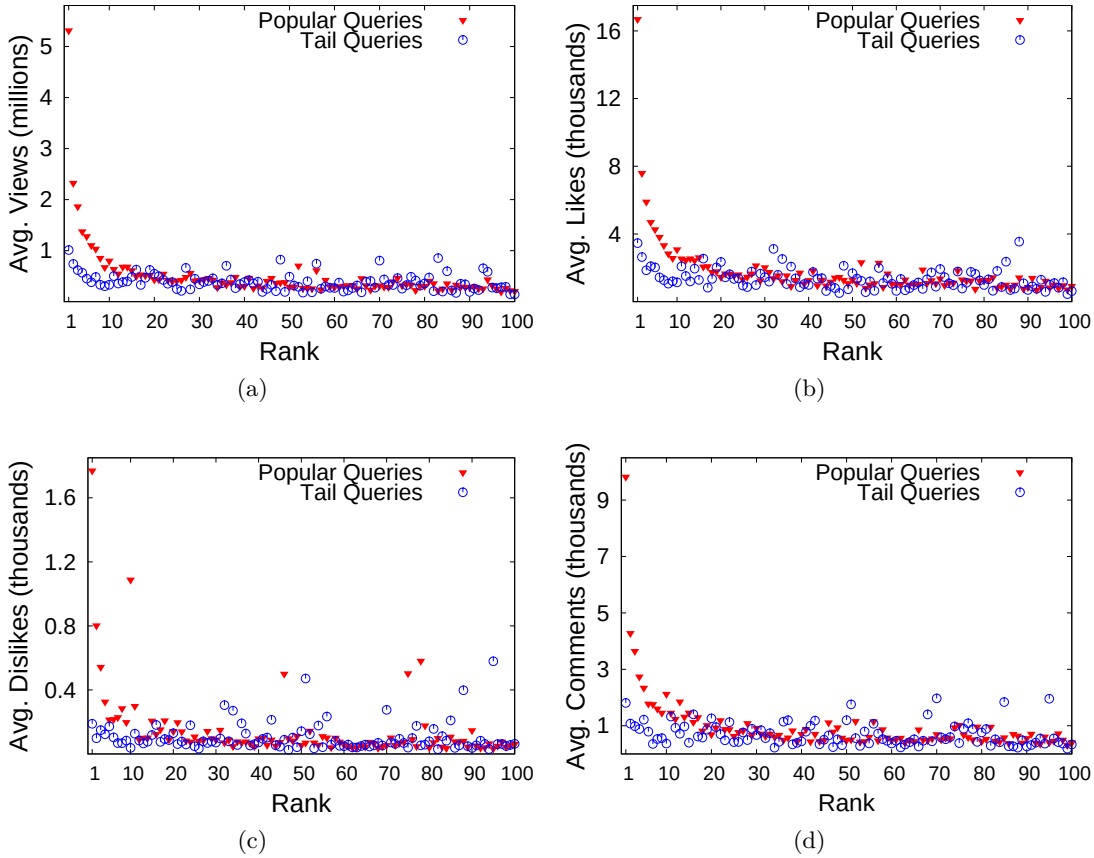


Figure 4.3: Avg. no. of (a) views, (b) likes, (c) dislikes and (d) comments vs. video rank in the query results (for the popular and tail queries).

Typically, YouTube users rate the viewed videos via clicking on the like and dislike buttons. We find that the videos in the top-10 (top-5) have higher number of likes and dislikes in comparison to the rest of the videos for the popular (tail) queries, respectively (see Figures 4.3b and 4.3c). Moreover, there is an order of magnitude difference between the number of likes and dislikes. For the popular queries, the average number of likes ranges from 16,689 (for the top video) to 916 (for the last video at rank 100), whereas the average number of dislikes is in the range of 1,769 to 62. For the tail queries, again the number of likes and dislikes at the corresponding ranks differ in an order of magnitude, but the variance for each kind of rating is smaller (e.g., the videos ranked at the first and last places have around 3,459 and 625 likes, respectively). An even stronger form of user interaction and participation in YouTube is posting comments for videos [120, 128]. Figure 4.3d depicts the average number of comments for videos at each result position. As in the previous cases, the top results have attracted considerably more attention than the others.

To summarize, we find out that the videos ranked in the first few results attracted

a very high user interest as expressed by the number of views, likes, and comments. The trend is observable for both query sets, but much stronger in the results of the popular queries. For the results ranked after the tenth position, the differences among the videos in terms of the values for these social features seem to be rather negligible.

On the one hand, the high values of the social features observed for the top-ranked videos can be attributed to the well-known Yule process (or rich-get-richer principle) [25], as the videos that appear in the first page and at higher ranks are more likely to be viewed and generate interaction. On the other hand, when all other features (such as the textual relevance to a query, etc.) are equal, it seems intuitive to place a video that is highly viewed/liked/commented etc. at a higher rank than the one that attracted no interest. In this sense, such social features are somewhat similar to the total click-count feature[4] that captures the global popularity of the web pages in typical learning to rank scenarios for the search engines. Furthermore, the number of ratings and comments provide a stronger evidence for the popularity than the number of views, as the latter can be more biased with the rank of the result while the former types of actions serve as the user's self-motivated feedback after viewing the content. Therefore, it seems as a promising direction to further investigate the retrieval potential of the social features in more depth.

4.3 Effectiveness and Correlation of Individual Features

In this section, we seek the answer for the following two questions: 1) How effective is each individual feature for ranking videos? and, 2) How are the rankings generated by the different pairs of features correlated? To answer these questions, for each query q, we need to re-rank the retrieved videos $v \in R_q$ (where $|R_q| \leq 100$) with respect to each feature f in our feature set, F. So, we begin with formally defining the basic and social features that are used for ranking the videos.

Basic features Basic features are based on the metadata fields created by the actual uploader of the video, namely, video title, tags and description. The features "title similarity" (f_{Title}), "tag similarity" (f_{Tags}) and "description similarity" (f_{Desc}) represent the vector-based similarity score of the query text, q, to a video's title ($TitleText(v)$), tags ($TagText(v)$), and description ($DescText(v)$), respectively. In our setup, for each of these metadata fields, we create the corresponding index using the Lucene 3.5 library[5] and employ its default retrieval function (based on the TF-IDF weighting model) to

[4] See http://research.microsoft.com/en-us/projects/mslr/feature.aspx
[5] http://lucene.apache.org

obtain the similarity scores for each (q, v) where $v \in R_q$. For each query and feature, Lucene scores are normalized to [0, 1] range.

Social features Social features are those that are formed due to some user interaction with the video *after* it becomes available. In this sense, as also discussed in the previous section, we first exploit the raw features provided by the system. For the metadata fields shown in Table 4.3, namely, number of views ($W(v)$), likes ($L(v)$), and comments ($C(v)$), we create the features f_W, f_L, and f_C, respectively. To be able to use them for ranking the videos, we normalize each feature value with the age of the video, $G(v)$. For the sake of completeness, we also consider the age of a video as a possible ranking feature and denote as f_G, while it is not a truly social feature (i.e., it is not based on the user interaction). Furthermore, we derive the following social features from the raw features and available data for our videos (all of these feature values are further normalized into [0, 1] range based on the maximum score observed for a given query):

- *Normalized no. of ratings (f_R)*: This feature represents the total number of ratings per video. The ranking criteria is: $(L(v) + D(v))/G(v)$.
- *Normalized ratio of likes (f_{RL})*: This feature captures the fraction of likes over all ratings for a video. The ranking criteria is: $(L(v)/(L(v) + D(v)))/G(v)$.
- *Normalized no. of comment authors (f_{CA})*: We extract the username fields from the crawled comments to capture the number of different users who commented on a video. The ranking criteria is: $A(v)/G(v)$ where $A(v)$ is the number of unique users who posted a comment for v.
- *Uploader popularity (f_{Up})*: The ranking criteria for a video v with an uploader u is $\sum W(v_j)/|Videos(u)|$, where $v_j \in Videos(u)$ and $Videos(u)$ includes the videos uploaded by u.
- *Comment similarity (f_{Com})*: We first aggregate the top-25 most popular comments (i.e., those with the highest number of likes) of each video into a single document and index these documents using Lucene. Then, the Lucene score between q and the comment document is computed for each $v \in R_q$.
- *Comment positivity (f_{Pos})*: We analyze the sentiment expressed in the comments by using SentiWordNet [53], as in [120]. Simply, this tool assigns a triplet representing the objectivity, negativity and positivity scores for each word in a comment, which are then averaged to obtain the overall scores for the comment. For ranking purposes, we only consider the average positivity score over all comments of a video, for which the tool can generate a score. The ranking criteria is:

$\sum Pos(c_i)/C(v)$ where $Pos(c_i)$ is the sentiment positivity score for the comment c_i of a video v.

- *Comment rating (f_{CR})*: We compute a comment's rating as the difference of the number of likes and dislikes that it has received. The ranking criteria for a video v is the average rating computed over all the comments posted for v.

- *Commenter popularity (f_{CP})*: We anticipate that the popular/active commenters would comment on the interesting and useful content. Therefore, for each unique commenter c of a video, we computed commenter popularity with the formula $\sum W(v_j)/|Videos(c)|$, where $v_j \in Videos(c)$ and $Videos(c)$ includes the videos uploaded by c. The ranking criteria for a video v is the average commenter popularity computed over all comments posted for v.

- *Commenter channel viewers (f_{CW})*: As another metric for the commenter popularity, we use the number of viewers for their YouTube channels. Again, we compute the average number of viewers of all unique commenters of a video as the ranking criteria.

- *Commenter channel subscribers (f_{CS})*: The ranking criteria for a video v is the average number of channel subscribers of all unique commenters of v.

- *Commenter contact (f_{CC})*: The ranking criteria is the average number of contacts of all unique commenters of a video.

In addition to the features listed above, we employ two more features that are created over the so-called *top-comments*, which are identified and displayed by YouTube among all comments of a video. In particular, we compute the values for Comment Rating and Comment Positivity only for these top-comments of a video, and refer to these features as Top Comment Rating (f_{TCR}) and Top Comment Positivity (f_{TPos}). Note that, these features are computed only for the videos of the tail queries, as top-comments are made available by YouTube only recently, i.e., after we have crawled all videos for the popular queries.

To sum up, our feature set F consists of three basic and seventeen social features in total, which are listed in Table 4.5 for easy reference.

User study To compute the retrieval effectiveness of each individual feature, we need the relevance annotations for all of the (q, v) pairs, where $v \in R_q$ and $|R_q| \leq 100$. As this task requires serious human effort, a subset of 50 queries is sampled uniformly at random from each one of our query sets, Q_P and Q_T. In order to obtain the relevance

Table 4.5: The list of all the basic and social features (F) employed in this work.

Notation	Description	Notation	Description
f_{Title}	Title-query Similarity	f_{Com}	Comment-query similarity
f_{Tags}	Tags-query Similarity	f_{CA}	No. of comment authors
f_{Desc}	Desc.-query Similarity	f_{Pos}	Comment positivity
f_W	No. of views	f_{CR}	Comment rating
f_L	No. of likes	f_{CP}	Commenter popularity
f_C	No. of comments	f_{CW}	Commenter channel viewers
f_G	Age	f_{CS}	Commenter channel subscribers
f_R	No. of ratings	f_{CC}	Commenter contacts
f_{RL}	Ratio of likes	f_{TCR}	Top comment rating
f_{Up}	Uploader popularity	f_{TPos}	Top comment positivity

judgments for these queries, we conducted a user study that involves 37 participants. Nine of the participants are female and the rest are males, and the age range is 20-35. All participants are from computer science related disciplines and 3 of them are undergraduates, 30 of them are graduate students, and the rest are post-docs. The participants are physically located in Germany, Turkey and USA.

We asked each participant to choose a few queries that are interesting for them from our set of queries. Each query is assigned to only one participant. We asked them to annotate the top-100 result videos for a given query using a 5-point rating scale, i.e., in the order of highly irrelevant, irrelevant, undecided, relevant, and highly relevant. Since videos are not downloaded but streamed directly from YouTube, it turned out that a small percentage of them have disappeared in time, i.e., been deleted by the uploader or not displayed in certain countries due to the copyright violation issues. The participants were asked to annotate such videos with rating 0. Finally, to avoid any bias, no social features were displayed along with the videos, but their titles and tags are kept to facilitate the judgment task.

Since a few queries retrieve less than 100 videos, we ended up with 4,969 and 4,949 relevance annotations for our popular and tail query sets, respectively. Table 4.6 shows the distribution of relevance labels for each query set. For the popular queries, 38% and 23% of the videos are judged as highly relevant and relevant, respectively; whereas only 15% of them are found irrelevant or highly irrelevant. In other words, more than half of the videos retrieved for the popular queries are judged to be relevant. In contrast, for the tail queries, only 27% of the videos are labeled as highly relevant or relevant, and 42% of the videos are found to be irrelevant or highly irrelevant. This implies that retrieving relevant videos for the tail queries is a harder task than that for popular queries, which is a rather expected result.

Note that, in the following experiments, we made a simplifying assumption and replaced 0 labels with the majority class for each query set (otherwise, for some rankings,

Table 4.6: Distribution of relevance labels.

		Popular		Tail	
Grade	Label	Count	%	Count	%
Not accessible	0	663	13%	293	6%
Highly Irrelevant	1	326	7%	1,769	36%
Irrelevant	2	414	8%	917	18%
Undecided	3	506	10%	633	13%
Relevant	4	1,143	23%	686	14%
Highly Relevant	5	1,884	38%	651	13%

we could end up with less than 10 videos, which is not preferable for the purposes of a fair comparison of the individual features). In particular, since most of the videos that were non-accessible for the popular queries were removed due to the copyright issues, i.e., implying that they were most probably the unpermitted copies of the official videos for the songs, movies, etc., we took an optimistic approach and set their labels as relevant. For the videos retrieved for the tail queries, for which case non-accessible videos were almost half of the number for the popular queries and there were no such copyright issues, we took a pessimistic approach and set their label as irrelevant, which is the majority class in this set. Note that, we also repeated the experiments reported in the rest of this chapter by completely discarding the videos with the rating 0, and found that all the trends remain the same.

Effectiveness of the individual features Our dataset presented in Section 4.2 allows us to compute all of the features described in the previous section except the ones that capture the commenter popularity, namely, f_{CP}, f_{CW}, f_{CS} and f_{CC}. For the latter set of social features, we further crawled the commenter profiles only for the annotated videos, as doing the same for all the videos would be very time-consuming due the access limitations of the YouTube. This yielded 23,721 and 57,181 commenter profiles for 4,969 and 4,949 annotated videos for the popular and tail queries, respectively. Then, for each query q, we obtained the top-10 ranking $R_{q,f}$ for each feature $f \in F$. In order to evaluate the performance of each individual ranking $R_{q,f}$, we computed the Normalized Discounted Cumulative Gain (NDCG) metric using the well-known `trec_eval` software package[6].

Figures 4.4a and 4.4b show the average $NDCG@10$ scores for each feature over all 50 queries from the popular and tail query sets, respectively. For the popular queries, the top-5 most effective features are f_{Tags}, f_{Desc}, f_{Title}, f_{Com} and f_G. For the tails, the order is slightly different, including f_{Title}, f_{Tags}, f_{Com}, f_{Desc}, and f_{TPos} in the top-5.

[6] http://trec.nist.gov/trec_eval/

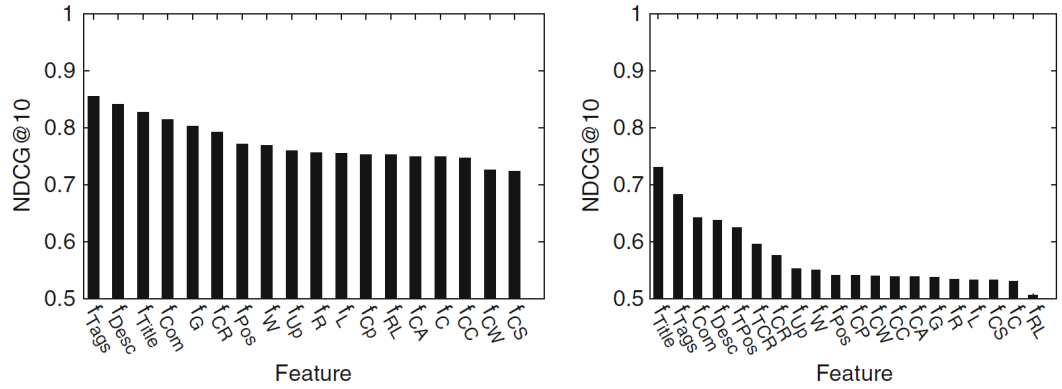


Figure 4.4: Average NDCG@10 for top-10 videos per feature for (a) popular, and (b) tail queries.

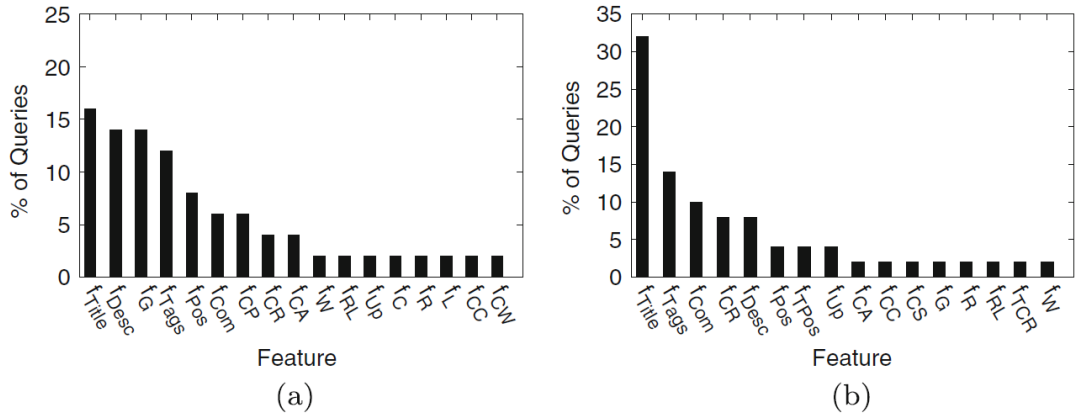


Figure 4.5: Fraction of queries for which a given feature yields the ranking with the highest NDCG@10 for (a) popular, and (b) tail queries.

The features derived from the comments seem to be the most promising social features, as they appear among the top-5 most effective features and perform comparable to the basic features for both query sets.

We explore how successful each feature is at a finer grain and compute the percentage of queries for which a particular feature yields the highest $NDCG@10$ score. Figure 4.5a shows that for the popular queries, the three basic features f_{Title}, f_{Desc}, and f_{Tags} provide the best rankings for 16%, 14% and 12% of the queries, respectively. This means that, the remaining 58% of the queries can benefit from the social features. We observe a similar situation also for the tail queries. Figure 4.5b reveals that the basic features provide the best results for only 54% of the queries and social features generate the best rankings for the rest, i.e., 46% of the queries.

In order to obtain the upper bounds for using basic and all features, we assume an oracle ensemble method choosing for each query the best ranking from the basic

features and all features (i.e., basic + social), respectively. In this idealistic case, the average NDCG scores would be 0.8935 (0.7688) for using only basic features, and 0.9261 (0.8149) for using all available features for the popular (tail) queries, respectively. This means that there is a non-trivial potential of improving the retrieval quality using social features in combination with the basic features within a machine learning setup, as we explore in Section 4.4.

Finally, we report the performance of the rankings as provided by YouTube, i.e., without any re-rankings. For our popular and tail query sets, YouTube achieves the average NDCG@10 scores of 0.8471 and 0.8035, respectively. These values are higher than the effectiveness of the individual features shown in Figure 4.4. This is expected given that YouTube has access to a much larger set of features, which are most likely to be utilized within a machine learning framework, as we also investigate in the rest of this chapter. Nevertheless, we are aware that our annotated datasets are too small to draw a general conclusion on the retrieval effectiveness of YouTube; so these figures are provided here only for the sake of completeness.

Correlation of the individual features We compute the pair-wise overlap between the features by averaging the similarity of their top-10 rankings $R_{q,f}$, $(f \in F)$ over all the queries in each query set, namely, Q_P and Q_T. A typical method for measuring the similarity of two ranked lists is using the Kendall's Tau metric [54]. Since we focus on the top-10 rankings, we employ a special variant of Kendall's Tau [54], which can handle the cases when a video appears in one of the top-10 rankings but not in the other.

In Table 4.7, we provide the Kendall's Tau scores that are normalized to [0-1] range where 0 means completely different rankings and 1 means equal rankings. The upper (lower) diagonal of the table presents the correlation of features based on the rankings for the popular (tail) queries, respectively. We find that, for the popular queries, the top-10 rankings provided by the feature pairs (f_L, f_R), (f_{CS}, f_{CW}), (f_{CC}, f_{CW}) and (f_{CS}, f_{CC}) are highly correlated, i.e., yield a similarity score higher than 0.85. Similarly, the top-10 rankings for the tail queries exhibit a high overlap, again greater than 0.85, for the feature pairs (f_L, f_R), (f_C, f_R), (f_C, f_L) and (f_{CS}, f_{CW}). These correlations between features imply some redundancy, at least for the purposes of ranking. In Section 4.4, we employ a feature selection approach to filter such redundant features.

4.4 Learning to Rank using Social Features

In the light of the above findings, it is promising to combine the basic and social features to optimize the retrieval performance. Moreover, as there is a high overlap between

Table 4.7: Kendall's Tau values for the feature pairs computed over the top-10 rankings for the popular (upper diagonal) and tail (lower diagonal, shaded) queries (The value 0 means completely different rankings and 1 means equal rankings). Note that, the features f_{TCR} and f_{TPos} are available only for the tail queries.

	f_W	f_L	f_{RL}	f_R	f_{Desc}	f_C	f_{CA}	f_{Title}	f_{Tags}	f_{Com}	f_G	f_{Up}	f_{Pos}	f_{CR}	f_{CP}	f_{CW}	f_{CS}	f_{CC}	f_{TCR}	f_{TPos}
f_W	1	0.75	0.30	0.76	0.07	0.72	0.59	0.17	0.11	0.08	0.26	0.47	0.04	0.11	0.12	0.10	0.10	0.10	N/A	N/A
f_L	0.82	1	0.45	0.98	0.08	0.82	0.69	0.17	0.10	0.07	0.38	0.40	0.03	0.09	0.10	0.11	0.10	0.09	N/A	N/A
f_{RL}	0.14	0.24	1	0.44	0.09	0.41	0.32	0.13	0.12	0.08	0.83	0.10	0.12	0.10	0.08	0.10	0.09	0.09	N/A	N/A
f_R	0.83	0.98	0.22	1	0.08	0.84	0.70	0.17	0.10	0.07	0.38	0.40	0.03	0.09	0.09	0.11	0.10	0.09	N/A	N/A
f_{Desc}	0.13	0.11	0.12	0.11	1	0.09	0.10	0.22	0.21	0.32	0.10	0.13	0.26	0.24	0.23	0.23	0.24	0.23	N/A	N/A
f_C	0.79	0.87	0.23	0.87	0.11	1	0.77	0.16	0.09	0.06	0.36	0.39	0.03	0.04	0.08	0.09	0.08	0.07	N/A	N/A
f_{CA}	0.70	0.76	0.20	0.76	0.11	0.83	1	0.14	0.09	0.07	0.27	0.33	0.04	0.05	0.09	0.11	0.09	0.08	N/A	N/A
f_{Title}	0.16	0.13	0.09	0.14	0.33	0.14	0.15	1	0.23	0.20	0.13	0.18	0.18	0.19	0.19	0.18	0.19	0.18	N/A	N/A
f_{Tags}	0.13	0.13	0.10	0.13	0.27	0.13	0.12	0.42	1	0.20	0.13	0.13	0.18	0.17	0.20	0.15	0.17	0.16	N/A	N/A
f_{Com}	0.27	0.25	0.08	0.25	0.19	0.25	0.27	0.23	0.24	1	0.13	0.11	0.56	0.55	0.53	0.52	0.54	0.54	N/A	N/A
f_G	0.12	0.20	0.83	0.22	0.12	0.20	0.17	0.11	0.12	0.08	1	0.07	0.15	0.19	0.17	0.18	0.17	0.17	N/A	N/A
f_{Up}	0.56	0.50	0.07	0.49	0.12	0.50	0.44	0.14	0.14	0.26	0.06	1	0.10	0.13	0.17	0.15	0.14	0.15	N/A	N/A
f_{Pos}	0.22	0.26	0.13	0.25	0.13	0.24	0.24	0.12	0.15	0.19	0.11	0.21	1	0.57	0.58	0.56	0.57	0.58	N/A	N/A
f_{CR}	0.29	0.30	0.08	0.29	0.11	0.25	0.22	0.16	0.13	0.18	0.07	0.23	0.21	1	0.75	0.75	0.76	0.76	N/A	N/A
f_{CP}	0.38	0.37	0.09	0.36	0.15	0.39	0.35	0.15	0.15	0.24	0.08	0.35	0.25	0.18	1	0.81	0.84	0.82	N/A	N/A
f_{CW}	0.32	0.32	0.09	0.32	0.14	0.35	0.34	0.16	0.17	0.24	0.08	0.27	0.28	0.17	0.65	1	0.93	0.87	N/A	N/A
f_{CS}	0.34	0.34	0.10	0.33	0.13	0.36	0.35	0.16	0.16	0.24	0.09	0.27	0.28	0.16	0.69	0.90	1	0.89	N/A	N/A
f_{CC}	0.34	0.33	0.10	0.33	0.13	0.35	0.34	0.16	0.15	0.22	0.07	0.27	0.27	0.17	0.61	0.81	0.78	1	N/A	N/A
f_{TCR}	0.39	0.37	0.06	0.36	0.14	0.36	0.32	0.19	0.18	0.22	0.06	0.32	0.23	0.52	0.29	0.28	0.26	0.26	1	N/A
f_{TPos}	0.27	0.26	0.09	0.25	0.18	0.26	0.24	0.29	0.23	022	0.08	0.26	0.25	0.32	0.24	0.21	0.20	0.22	0.55	1

the rankings generated by certain pairs of the features, it also seems reasonable to apply a feature selection algorithm. In what follows, we present our video retrieval framework involving a number of state-of-the-art learning to rank (LETOR) strategies and a greedy feature selection strategy adapted from [60]. In this framework, we explore the impact of social features on the video retrieval.

4.4.1 Video Retrieval Framework

The LETOR algorithms proposed in the literature fall into three categories, namely, point-wise, pair-wise and list-wise [27]. In this thesis, we employ seven LETOR approaches that cover all of these categories. We provide a concise description of each approach in Section 2.4. For all of the algorithms, we experiment with various parameter values and report the results for the best-performing configuration for each setup. We also specify when these best-performing configurations differ for the popular and tail query sets. In particular, for *RankSVM*, the trade-off parameter between the training error and margin is set to 10. For *RankBoost*, the number of rounds for training is set to 300 (50) and the number of threshold candidates to search is set to 5 (10) for the popular and tail query sets, respectively. The number of training epochs is set to 500 for *ListNet*. For *CoordinateAscent*, for both query sets, the number of random restarts is 5 and number of iterations to search in each dimension is 10. We also set the metric to optimize on training data as the NDCG. For *GBRT*, *RF* and *iGBRT*, we use the same parameters for both of the query sets. For *GBRT*, the tree depth parameter is set to 2, number of trees for the ensemble is 1,000, and learning rate is 0.1 (the latter two values are also used in [98]). For *RF*, again following the practices in [98], we set the tree depth as 10% of the number of features (i.e., this is 2 in our setup, as we have at most 20 features) and number of trees as 10,000 since the algorithm is safe for not to overfit. Finally, for *iGBRT*, we used the above parameters for *RF*, obtained predictions for the current test set and piped these predictions to *GBRT* that is also invoked with the above parameters for the original algorithm.

Feature selection for LETOR approaches Feature selection is a popular strategy in machine learning for enhancing the accuracy of the learned model [60, 42]. In what follows, we discuss two greedy feature selection strategies to address the optimization problem for feature selection. First, we briefly review a strategy, so-called GAS (Greedy search Algorithm of Feature Selection), that is introduced by Geng et al. [60]. Next, we propose to adopt a well-known strategy, Maximal Marginal Relevance [23], for the feature selection in our learning to rank framework.

- *GAS:* This is a greedy search strategy [60] that starts with choosing the feature, say f_i, with the highest importance score $Imp(f)$ into the top-k feature set, S. Next, for each of the remaining features f_j, the importance score is updated with respect to the following equation:

$$Imp(f_j) \leftarrow Imp(f_j) - Sim(f_i, f_j) \cdot 2c, \tag{4.1}$$

  where c is a constant to balance the importance and similarity optimization objectives. The algorithm proceeds with choosing the next feature with the highest importance score and updating the remaining scores, until k features are determined.

- *MMR:* This is again a well-known greedy strategy [23] that is originally introduced for the search result diversification problem; i.e., to construct both relevant and diverse top-k results for a given query. We adopt MMR to choose the features that yield both the highest average effectiveness and, at the same time, the most diverse rankings. In a similar manner to GAS, the MMR strategy also starts with choosing the feature f_i with the highest importance score into the top-k feature set, S. Next, in each iteration, MMR computes the score of an unselected feature f_j according to the following equation:

$$Score(f_j) \leftarrow c \cdot Imp(f_j) - (1 - c) \max_{f_i \in S} Sim(f_i, f_j), \tag{4.2}$$

  where c is again a constant to balance the importance and similarity. In other words, the score of f_j in MMR is computed by discounting the feature's importance score with its maximum similarity to the features that are already selected into S.

In our case, following the practice in [60], the feature importance score, $Imp(f)$, is set to the $NDCG@10$ score of f obtained over the queries in the training set. The similarity score $Sim(f_i, f_j)$ between any two features f_i and f_j is computed by the variant of the Kendall's Tau metric (described in Section 4.3) between their top-10 rankings, again, over the queries in the training set.

4.4.2 Experimental Results for Feature Selection

In our LETOR framework, all experiments are conducted using five-fold cross validation over the popular and tail query sets (with 50 queries in each) as described in the previous

section. For each fold, we first used the training set of 40 queries (i.e., around 4,000 annotations) to determine the k-feature sets (where $1 \leq k \leq 20$) from the set of all basic and social features[7], i.e, F using the greedy selection algorithm. Next, for each value of k, the LETOR algorithms in our repository are trained with the same set of instances and these k features; and tested on the remaining 10 queries (around 1,000 annotated instances). The average $NDCG@5$ and $NDCG@10$ scores are computed using `trec_eval` software for the test queries. The final scores are obtained by averaging over the folds.

In Figures 4.6 and 4.7, we provide the performance of each LETOR algorithm with respect to the number of features selected with GAS or MMR strategies, for the popular and tail query sets, respectively. As in [60], the performance fluctuates as the feature set grows. Nevertheless, for almost all cases, there exists a set of features, so-called the *best-k* set, that yields a higher performance than using all of the features, which justify our use of a feature selection algorithm. We further observe that the MMR strategy, that is adopted within the LETOR framework in this chapter, is comparable to GAS, and for particular cases, it can even outperform the latter.

4.4.3 Experimental Results for the Impact of Social Features

To expose the potential of social features for video ranking, we compare the retrieval performance of using the best-k feature sets (from Figures 4.6 and 4.7) to the performance of using only the basic features. For the latter case, we employ the features f_{Tags}, f_{Title} and f_{Desc} for training and testing all of the LETOR algorithms. Our findings are reported in Table 4.8. Note that, all of the best-k sets that were obtained with the GAS strategy include some social features as k is always found to be greater than the number of basic features, i.e., 3. The findings are similar for the majority of the cases with MMR.

Before discussing our results, please note that we avoid comparing the LETOR algorithms to each other in our framework. This is because the one-way ANOVA test for comparing the $NDCG@10$ scores of 50 queries for these six algorithms reveals that the performance differences among them are usually not significant, regardless of the feature set they employ (i.e., the basic or best-k features). In other words, it is not accurate, from statistical point of view, which of these algorithms performs best in our video retrieval scenario, and thus it is important to improve the performance of any of these algorithms using the social features.

As Table 4.8 reveals, for all the algorithms, the best-k features can improve the

[7]Recall that two of the features, f_{TCR} and f_{TPos}, were not available when we crawled the data for the popular queries and hence, at most 18 features can be used for these queries.

Table 4.8: Average $NDCG$@10 scores for LETOR algorithms using the basic and best-k features obtained with the GAS and MMR strategies for the popular and tail query sets (for bold cases, differences from the baseline are statistically significant). For GAS and MMR, we also denote the number of selected features (k) in parentheses.

	RankSvm	RankBoost	CoordAsc	ListNet	GBRT	RF	iGBRT
				Popular Queries			
Basic features	0.8655	0.8092	0.8356	0.8243	0.8528	**0.8073**	**0.8073**
GAS	0.8664 (2)	0.8228 (6)	0.8425 (4)	0.8378 (16)	0.8616 (6)	**0.8547** (9)	**0.8605** (9)
MMR	0.8691 (2)	0.8146 (6)	0.8424 (7)	0.8384 (14)	0.8581 (2)	**0.8588** (12)	**0.8576** (18)
				Tail Queries			
Basic features	0.7356	0.7284	0.7204	0.7197	0.7370	**0.6715**	**0.6681**
GAS	0.7537 (4)	0.7545 (8)	0.7428 (11)	0.7336 (7)	0.7437 (10)	**0.7258** (10)	**0.7272** (10)
MMR	0.7506 (10)	0.7549 (8)	0.7428 (9)	0.7332 (5)	0.7446 (15)	**0.7332** (10)	**0.7345** (13)

$NDCG$@10 scores that are obtained by the basic features alone. For some cases, the improvement is numerically small, i.e., around 1% (though, so are most of the results reported in the LETOR literature, e.g., see [27, 98]) while in some other cases, using particular social features in combination with the basic features can add up to an absolute 7% to the effectiveness. The gains in $NDCG$@10 obtained by using the best-k sets with social features are found to be statistically significant on a 95% confidence level for the RF and iGBRT approaches. The smaller numeric improvements in $NDCG$@10 scores, observed for the other algorithms, are not statistically significant.

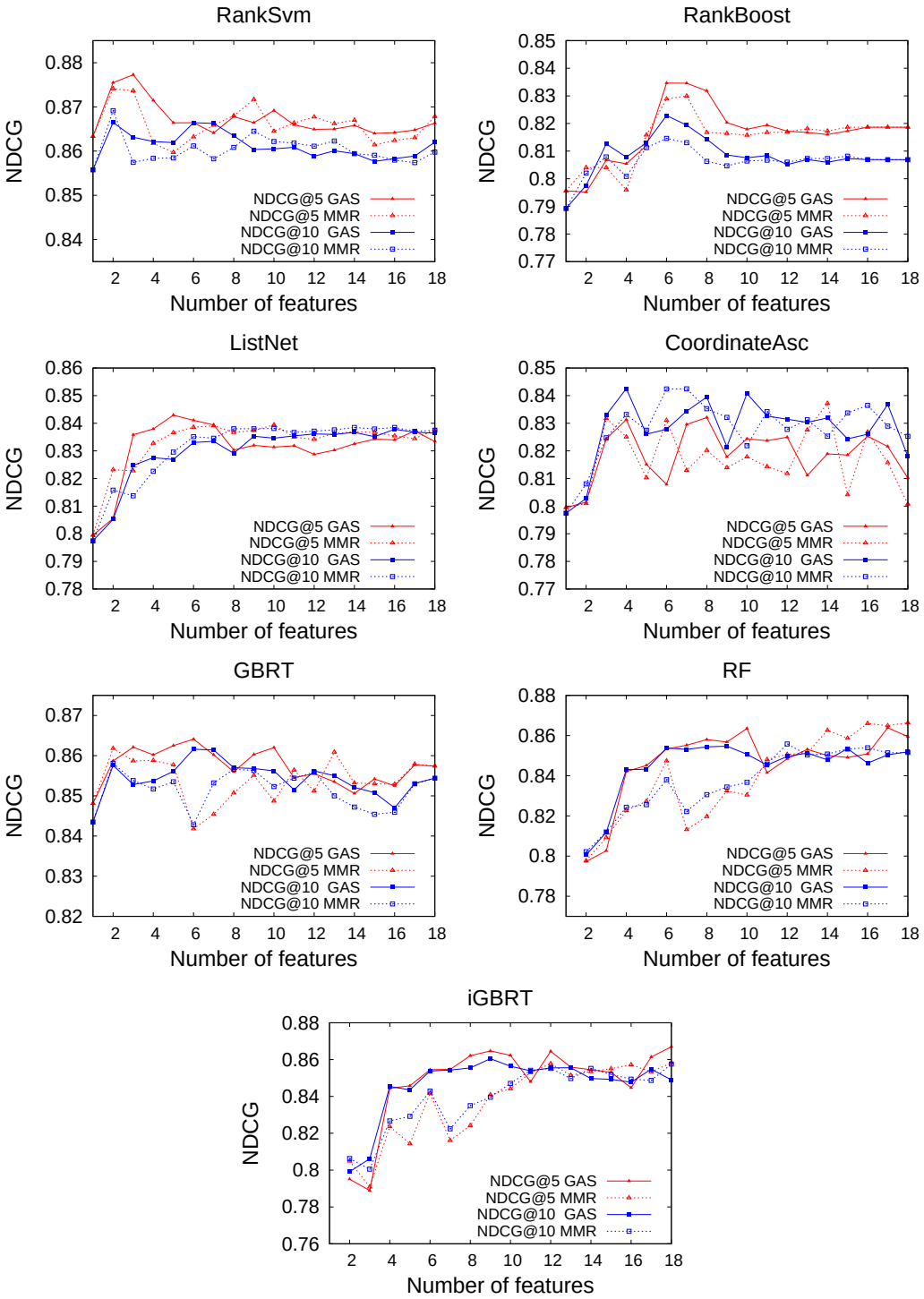


Figure 4.6: *NDCG* scores for the LETOR algorithms w.r.t. the number of features for the popular queries.

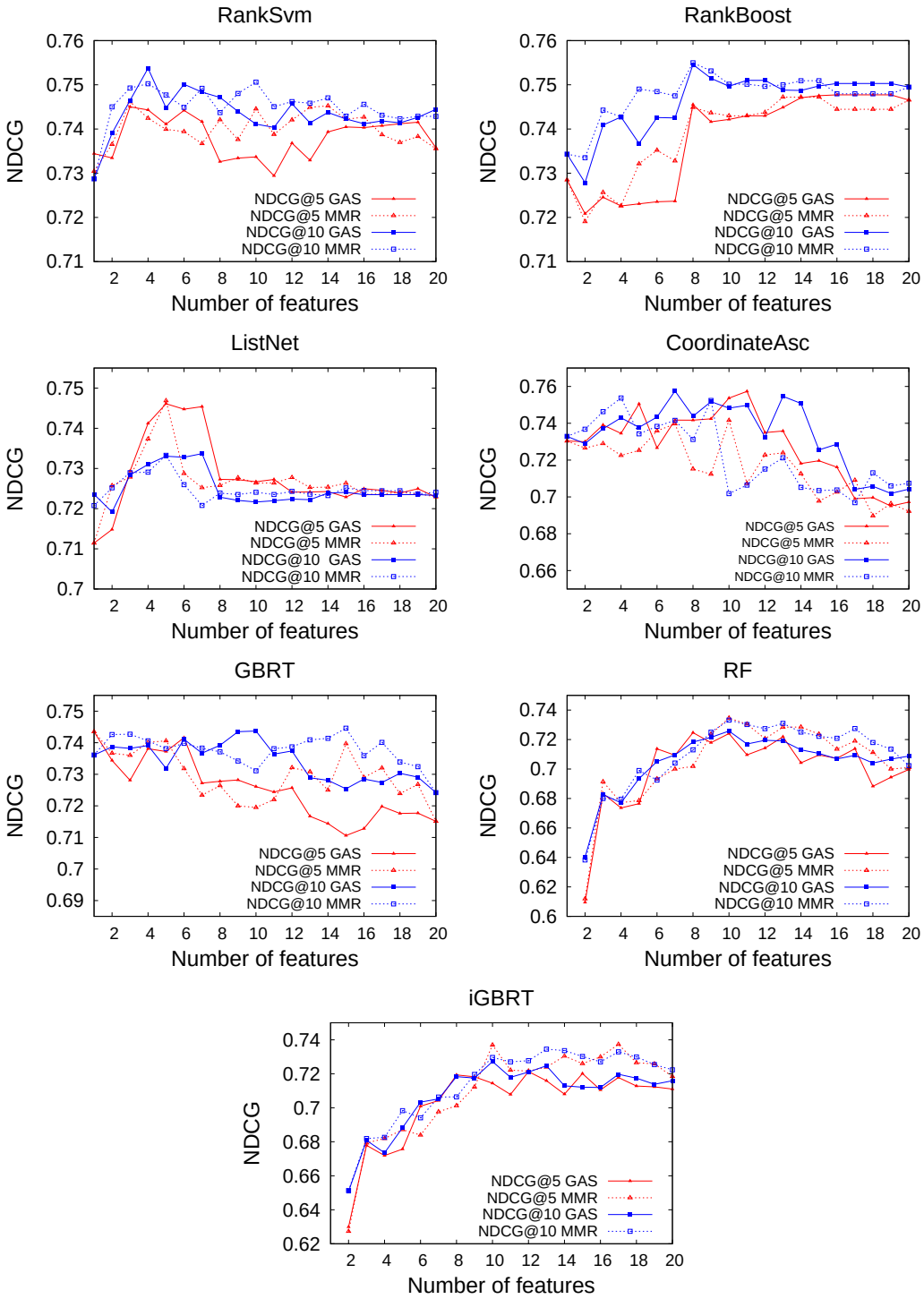


Figure 4.7: *NDCG* scores for the LETOR algorithms w.r.t. the number of features for the tail queries.

4.5 Summary and Contributions

Our key contributions in this chapter are as follows.

We show that the social features can improve the video retrieval performance when combined with the basic features, which indeed constitute a very strong baseline on their own. Furthermore, we demonstrate the usefulness of such features not only for the popular queries, for which there might be additional clues obtained from the abundant click data, but also for the tail queries, for which such click information is very scarce. This latter finding is worthwhile, given that the competition among search engines is becoming more focused on queries in the long tail (e.g., [148]).

Our experiments reveal that using all the basic and social features within a LETOR framework is ineffective, and feature selection strategies can successfully eliminate the redundant features (i.e., those that have low retrieval effectiveness and/or high overlap with the already selected features). In contrast, the best-k sets still include several social features (see the values of k in Table 4.8), which indicates that some social features that were not so effective on their own (as shown in Section 4.3) turn out to be useful when combined with the other basic and social features.

We finally show that the MMR strategy, as we adopt in this chapter, is comparable or even superior to GAS for the purposes of feature selection.

5 Community Sentiment in Web Queries

In this chapter we present an in-depth analysis of Web search queries for controversial topics, focusing on query sentiment. To this end, we conduct extensive user assessments and discriminative term analyses, as well as a sentiment analysis using the SentiWordNet thesaurus. Furthermore, in order to detect the sentiment expressed in queries, we build different classifiers based on query texts, query result titles, and snippets. We demonstrate the virtue of query sentiment detection in two different use cases. First, we define a query recommendation scenario that employs sentiment detection of results to recommend additional queries for polarized queries issued by search engine users. The second application scenario is controversial topic discovery, where query sentiment classifiers are employed to discover previously unknown topics that trigger both highly positive and negative opinions among the users of a search engine.

5.1 Related Work

There is a plethora of work on *sentiment classification, opinion mining, and opinion retrieval* [103]. Recent work in this area makes use of annotated lexical resources such as SentiWordNet [53] or SentiStrength [127] to improve classification performance. Several other studies make use of sentiment thesauri for exploratory studies. In [120], we analyze the connection between sentiment in comments and community ratings using the SentiWordNet thesaurus.. In [85] the SentiStrength thesaurus is leveraged for studying sentiment in Yahoo! Answers with respect to temporal and demographic aspects. In [138], the same sentiment thesaurus is used to guide a focused crawler for discovering opinionated web content. In this chapter we apply sentiment analysis in what is a novel context, i.e., Web search queries.

In [44] the authors make use of sentiment analysis to compare the sentiment ex-

pressed in query *results* (in contrast to the queries themselves) for different search engines. Pera et al. [107] suggest an approach for summarizing query results with respect to sentiments and facets. [140] analyze aspects like topics, trends, and opposition in search results for left and right leaning queries related to US politics. The political polarity of queries (left or right) is determined by click behavior on left or right wing blogs. To our knowledge, the work closest to our study is a recent paper by Gyllstrom and Moens [67]. Here, the authors also point out that a number of Web queries represent an opinion; however, they use such queries to detect controversy of a *given* topic in a search engine for children, so that additional protective mechanisms can be triggered if children search for such topics. To decide on the controversy of a topic, they obtain a set of suggestions for a given topic from a major search engine, and then create the negations of these suggestions (using antonyms and negating terms). If such queries with negations (i.e., anti-queries) also appear in the suggestion list of the search engine, the given topic is considered as controversial.

This chapter also has some connections to the well-known concept of *semantic markedness* in the linguistics literature, which suggests that for a certain pair of related words, one can be unmarked whereas the other can have a semantic orientation/implication. As exemplified in [71], for the adjective pair tall-short, the term "tall" is the semantically unmarked one as there is no implication in the question "How tall is Jack?", whereas replacing "tall" with "short" in the question would imply that the speaker thinks that Jack is indeed short. We anticipate that findings from this area (e.g., see [71, 72]) can be applied for analyzing and/or detecting the sentiment in queries, which is an interesting future work direction.

Controversy has also been studied for data sources other than queries. [84] analyzes conflicts in Wikipedia updates, and apply machine learning techniques using characteristics of the update history as features in order to predict articles containing controversies. In [137] articles in Wikipedia are ranked by controversy using information about the conflicting interaction between contributors. In [8] the OpinionNetIt system is proposed for extracting opinion holders (e.g. politicians) and their opinions about different topics and facets; the resulting information can be leveraged to detect controversies. Data sources used for information extraction include Google News, Wikipedia, and websites of newspapers. However, we are the first to explore controversy of opinions in the context of query analysis.

In the context of *regional differences between queries* Rogers et al.[1] explore different local Google versions for determining which specific types of rights (e.g. children's rights, patients' rights) are most frequently searched for in different countries. However,

[1] https://wiki.digitalmethods.net/Dmi/NationalityofIssues

they do not analyze the sentiment expressed in queries.

There is a considerable amount of work on *classification of queries* into different taxonomies. Taxonomies can be based on the general user intention such as the Transactional, Navigational and Informational query intents introduced by Broder [16] or can be topic-oriented (e.g. "Entertainment" vs. "Sports" vs. "Politics" as well as subcategories). In [18] queries and result documents are used to build language models, and queries are classified into categories of a topic-based taxonomy taken from a leading search engine and consisting of several thousand nodes. In [82] queries are classified as being related to a "homepage finding task" or a "topic relevance task", exploiting information contained in query terms, part-of-speech information in queries, and terms from anchor texts and titles. In addition, there is work on leveraging click graphs and query sessions in order to propagate query category labels [87], and for category label disambiguation [21]. In [117] an approach for classifying queries into a product taxonomy is suggested. However, none of these works studies *sentiment* connected to queries.

Query recommendation aims at suggesting additional relevant queries for a given query. Baeza-Yates et al. [9], for instance, use a combination of term-based query similarity and query support (obtained through the number of document clicks for queries) to suggest relevant queries. Fonseca et al. [57] mine association rules from sets of query sessions in order to identify related queries. A template-based approach for mining recommendation rules involving general entity types such as city, person, or substance is described in [125]. In [7] transition probabilities inferred from subsequent queries in user sessions are leveraged for query recommendation. In contrast to these works, we make use of sentiment-based query relationships in order to recommend queries that are aligned with the sentiment expressed in the original query.

5.2 Data Gathering, Methods and Characteristics

5.2.1 Data Collection

In order to obtain opinionated queries on controversial topics, we gathered a set of 50 such topics from three different resources. First, we used all of the 14 controversial topics that were employed in [44] in the (different yet related) context of investigating the sentiment in the *search results* retrieved by different search engines. In addition, we sampled 36 topics from the Web sites `http://www.procon.org/` and `http://en.wikipedia.org/wiki/Wikipedia:List_of_controversial_issues`. The former source is a non-profit and popular Web site that is referred to by various educational institutions as an online resource. The latter site includes the list of controversial

Table 5.1: List of controversial topics (along with the number of manually annotated queries per topic).

Topic	Queries	Topic	Queries	Topic	Queries
abortion	189	euthanasia	188	john mccain	124
anorexia	185	fidel castro	117	judaism	179
barack obama	129	gaddafi	116	marijuana	189
bill clinton	172	gay marriage	129	marriage	185
bit torrent	79	genetic engineering	118	nato	179
britney spears	186	george bush	188	nuclear energy	127
bullfighting	114	global warming	123	nuclear power	126
christianity	187	gypsies	157	obesity	185
circumcision	107	hamas	120	patriotism	130
climate change	129	hillary clinton	183	prostitution	184
cloning	186	hippies	118	ronald reagan	119
communism	187	homosexuality	186	sarah palin	123
cyprus	183	hugo chavez	105	scientology	187
death penalty	187	human cloning	117	stem cell research	120
drinking age	124	immigration	188	terrorism	188
economy	189	iphone	128	vegetarianism	119
employment	181	islam	192		

topics that caused a large number of edit conflicts in the corresponding Wikipedia articles, and has been employed in other studies as well (e.g., see [67]). From these two Web sites, we discarded topic names that are unlikely to be issued as a keyword query (e.g. the topic name "prescription drug ads to consumers" from `procon.org`). For all selected topics, we used the topic name (as-is) as initial query. The resulting list of topics is shown in Table 5.1.

Ideally, we would consider all queries submitted to a search engine on a particular topic, in order to determine the fraction of opinionated queries and analyze sentiment in them. As query logs are precious assets of search engines and usually kept confidential, instead, we opted for exploiting publicly available resources to sample an adequate number of queries. To this end, for each topic, we gathered a set of queries using a major search engine's query suggestion (auto-completion) service and the AOL query log [106].

For obtaining queries from the auto-completion service, we created 5 different templates, as listed in Table 5.2. With the first, most general, template we collected all query auto-completions as ⟨*topic*⟩ followed by any letter in the English alphabet (e.g., when typing the query "abortion a", suggestions are "abortion articles", "abortion arguments", etc.). These instant suggestions are usually constructed from the popular and related queries submitted by other users [119, 12]. Note that, if the prefix terms in a query do not fully match to any of the queries in the suggestion database, some search engines still auto-complete only the last term being typed. Therefore we did not

collect this type of partial suggestions. This guarantees that our dataset includes only *full* queries that were actually issued by users. The Google web search help page[2], for instance, states that these auto-completions are "a reflection of the search activity of all web users and the content of web pages indexed by Google."

Our second template just prefixes the topic name with one of the six interrogative pronouns (who, where, what, when, why, how) and an appropriate auxiliary verb (e.g., typing "why is abortion" returns suggestions like "why is abortion bad/good/legal", etc.). The remaining three templates use auxiliary verbs "is/are" (with negations) and, as also discussed in [67], are more biased towards opinionated queries. However, via template 5, we again obtain all queries that are in the form of "$\langle topic \rangle$ is [*letter*]", yielding a more general set than the one in [67]. Finally, we also selected all queries from the AOL query log containing the topic name. The AOL log includes around 20M queries submitted to the AOL search engine in 2006. Since almost half of the topics match very few or no queries in this log (due to limited size and older date of this log), we only gathered queries for 26 of the topics. This case is denoted as template 6 in Table 5.2. Note that, we intentionally chose generic templates (rather than introducing additional bias in templates to find a larger number of opinionated queries). We did this in order to get a better idea of the role of sentiment in real-world query streams. The overall process yielded a total of 31,053 queries for our 50 topics. For each of these queries, the top-10 query result titles, URLs, and snippets were gathered using the Yahoo! Search API (in June 2011).

While we attempted to use as much of this dataset as possible in our studies, for some analyses or experiments the amount of annotated data and number of redundant annotations we could gather varied due to the relatively high costs and varying availability of human judges. However, we describe the setup and number of judges involved in each individual experiment, and we made sure that data samples were chosen uniformly at random (unless specified otherwise).

5.2.2 Characteristics of Opinionated Queries

5.2.3 Sentiment in Web Search Queries

We first investigated the sentiment expressed in queries by conducting an annotation study.

Setup We randomly sampled sets of queries proportional to the total number of queries in the templates. For each topic, we asked users to annotate around 130 queries obtained from search engine suggestions, and an additional set of 60 queries from the

[2] http://support.google.com/websearch/answer/106230?hl=en

Table 5.2: Templates for gathering queries (along with the number of manually annotated queries per template): queries for templates 1-5 are obtained using the query suggestion service, and those for template 6 are extracted from the AOL log.

Id	Template	Queries
1	`<topic> [letter]`	2,664
2	`what, why, how, where, who, when (is|are) <topic>`	300
3	`<topic> (is|are) [blank]`	345
4	`<topic> (is|are) not [blank]`	349
5	`<topic> (is|are) [letter]`	2,458
6	`[letter]* <topic> [letter]*`	1,535

AOL log (if available). The annotator pool included undergraduates, PhD students, and Post-docs in the area of computer science. Each user was assigned at most three different topics, and asked to label a given query as positive, negative or neutral, depending on the opinion expressed in the query text. Furthermore, the queries for five randomly chosen topics (corresponding to 929 queries) were separately annotated by two of the authors of our published paper [30] in order to get an idea of the inter-user agreement. Fleiss' Kappa [66] was found to be 0.7. Note that according to Fleiss' definition, $\kappa < 0$ corresponds to no agreement, $\kappa = 0$ to agreement by chance, and $0 < \kappa \leq 1$ to agreement beyond chance. Due to the high inter-agreement and the large number of queries to be annotated, queries for each topic were assigned to only one annotator.

Results Overall we obtained a set of 7,651 annotated queries (see Tables 5.1 and 5.2 for the breakdown of annotated queries per topic and template, respectively), with 890 and 1,490 of them annotated as positive and negative, respectively, and the rest as objective. Figure 5.1 shows the distribution of queries from each template to one of the three sentiment classes. As might be expected, queries from the first (most general) and last (AOL log) templates represent a random sample for a given topic, and are dominated by objective queries (around 90%). Still, there remain a non-negligible 10% of opinionated queries. The queries in the form of questions exhibit a slightly larger fraction of polarization, with the percentage of objective queries dropping to 85%. The templates of the form "$\langle topic \rangle$ is/are ..." naturally reveal the highest subjectivity, with 67% of queries in one of these forms involving an opinion expressed in the query text. Table 5.3 shows a couple of example queries for each sentiment class for the topic "George Bush".

Term Analysis We also conducted a term analysis on the query texts to compare the objective vs. subjective and positive vs. negative classes. For each case, we ranked

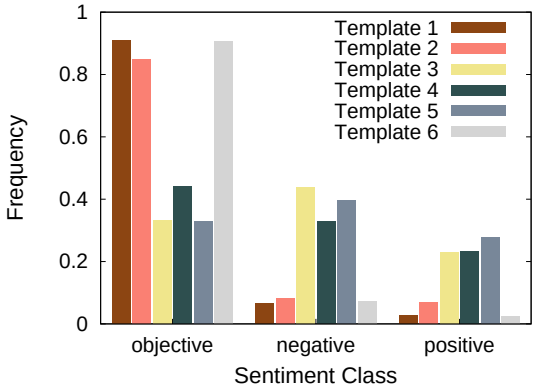


Figure 5.1: Distribution of queries over the sentiment classes for different templates.

Table 5.3: Queries and sentiment categories for the topic "George Bush".

Objective	**Positive**	**Negative**
mr george bush birthday	george bush is smart	george bush is the worst president
george bush is from texas	george bush is my hero	george bush is a lizard
george bush oil	george bush is awesome	george bush is incompetent
pre iraqi war speech george bush	george bush is not that bad	george bush doesn't care about black people

the query terms (after stemming) using the Mutual Information (MI) measure [92], which essentially quantifies how much the joint distribution of terms deviates from a hypothetical distribution in which terms and sentiment classes are independent of each other. In the literature, Pointwise Mutual Information is also employed in an unsupervised method for detecting sentimental words [132, 133]; however in this work we use MI only for identifying the most distinctive terms of the queries for each sentiment class. Table 5.4 shows the top-20 (stemmed) terms with highest MI scores for the objective vs. subjective (left) and positive vs. negative classes (right). Note that term lists are quite intuitive in that the objective class involves mainly general query terms (e.g. *fact*, *question*, *journal*) whereas most of the subjective terms express some opinion (e.g. *racist*, *worst*, *stupid*). A clear distinction between positive and negative query terms can also be observed in Table 5.4.

Recall that in the user study, a considerably larger number of queries was labeled as negative rather than positive (i.e., 1,490 vs. 890). This is also reflected by the terms shown in the second column of Table 5.4, as most of the subjective terms seem to convey a negative feeling. Interestingly, our recent work on user comment analysis reports that users in social communities tend to cast more positive than negative votes (see [120]). A possible reason for this (rather contradictory) finding might be that, in case of the Web search activity (which is an individual act rather than a social one), users express more negative feelings, maybe for the purposes of finding like-minded

Table 5.4: Top-20 (stemmed) query terms w.r.t. MI values for objective vs. subjective category (left) and positive vs. negative category (right).

Terms for Objective Queries		**Terms for Subjective Queries**		**Terms for Positive Queries**		**Terms for Negative Queries**	
com	issu	bad	moral	good	life	bad	child
new	use	good	gay	right	awesom	wrong	wors
vs	state	wrong	crime	legal	posit	kill	problem
www	doe	right	idiot	import	best	evil	sin
countri	pictur	kill	racist	great	hero	gay	failur
lyric	question	evil	diseas	better	pro	danger	dumb
statist	japan	stupid	uneth	cool	funni	dead	retard
fact	journal	problem	great	moral	world	worst	old
2011	yahoo	hero	safe	healthi	futur	uneth	stupid
histori	york	danger	worst	hot	religion	racist	hitler

people complaining or providing solutions for the same issue.

5.2.4 Analysis of Query Volumes

We also conducted an analysis of the volume of queries containing sentiment; in particular we studied the frequency of queries from the different sentiment classes, using Google's Keyword Tool[3]. This service was created to help choosing appropriate ad words and provides the local and global monthly average search volume of a query over the last 12 months for the selected countries, languages and devices (e.g. desktops, laptops, or mobile devices). We chose not to use Google Trends[4], because, although employed in recent works (e.g. [109]), it only provides relative volume information and, thus, cannot be used for comparing and aggregating volumes across different queries.

Setup Since Google Keyword does not allow for submitting a large number of queries automatically, we employed a crowdsourcing solution. To this end, for each query, we created a Human Intelligence Task (HIT) in Amazon Mechanical Turk (AMT) that asked workers to submit a given query to the Google Keyword Tool and to provide the returned volume (or, "-1" if no volume information was found). To study the agreement among workers for this task, we assigned all of the 378 queries for two of our topics, namely "abortion" and "euthanasia" to two different AMT workers. Crowdsourcing provided very reliable results for this task: the volume values entered to HITs differed only for two of the queries. Due to the very high overlap among the annotators, the remaining queries were assigned to only one AMT worker.

[3] https://adwords.google.com/o/KeywordTool
[4] http://www.google.com/trends/

Results Among the 7,651 annotated queries in out dataset, the Keyword Tool did not provide a volume value for 3,256 (42.54%) of the queries. This might be due to the time gap between constructing our query set and using the Keyword Tool for collecting volumes, or differences in the procedure used for generating query auto-completions and volume values by the search engine. Nevertheless, there is no significant bias towards a particular sentiment class; we found that 41%, 54%, and 43% of the queries labeled as objective, positive, and negative, respectively, had no associated volume. For the remaining 4,392 queries, we observed a typical heavy-tailed distribution of query frequencies. The total volume adds up to 257,751,783, with objective queries amounting to 97% of the volume. However, a non-negligible amount of 3% of the query volume (i.e., around 7.5 million queries per month) for our topics are opinionated. The imbalance between negative and positive queries (as shown in the previous section) is still apparent yet less strong, with each class containing 4,319,345 and 3,208,531 queries, respectively.

5.2.5 Sentiment in Query Results

In addition to analyzing the sentiment in query texts, we also investigated the traces of bias in the query results. In their previous study, Demartini and Siersdorfer employed an automatic approach to investigate the opinions expressed in top-ranked results of controversial topics [44]. In that study, different from our work, the initial query (i.e., simply the topic name) is not opinionated, and the goal was to analyze the search result lists for queries on controversial issues. The authors report that on average, the top results returned from three different search engines do not express extreme opinions. We complement and extend that study by providing a manual analysis on the results of opinionated queries in the rest of this section. We discuss the lexicon-based analysis of opinions in queries in Section 5.2.7.

Setup For our study, we randomly selected three queries labeled as objective, positive and negative for 20 of our topics (again chosen uniformly at random) during the annotation process. For each query, we retrieved the top-10 query result titles and snippets via the Yahoo! API. Next, we shuffled these query results and annotated each title and snippet as positive, negative, or objective. In this way we obtained 600 annotated titles and the same number of annotated snippets.

Results Figures 5.2(a) and (b) show the distribution of sentiments in result titles and snippets, respectively, for each query class. Our experiment reveals that, regardless of the opinion in the query, most search results do not express a considerable bias towards

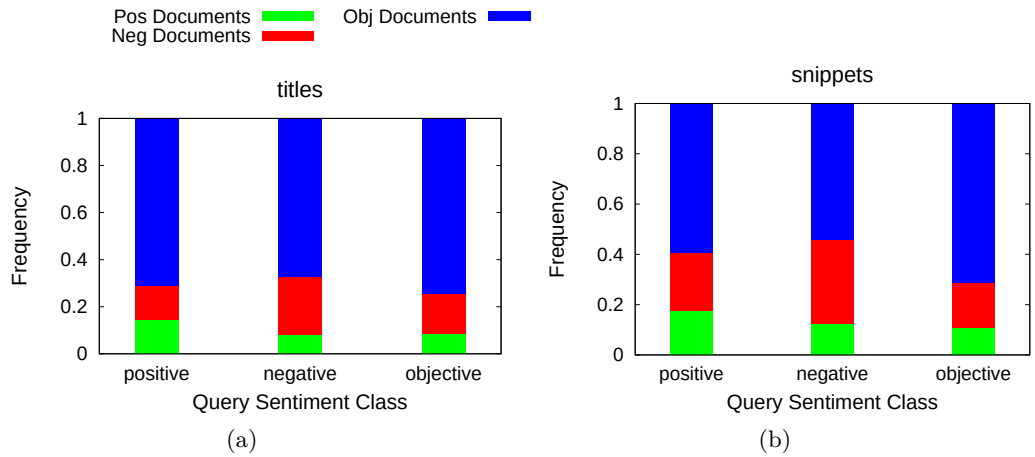


Figure 5.2: Sentiment distribution of (a) query result titles, and (b) query result snippets for the queries from each sentiment class.

an opinion. Titles, which are shorter, are found to be more objective than snippets. On the other hand, the fraction of positive (negative) results is larger for positive (negative) queries than for the other queries. For instance, the fraction of negatively labeled snippets retrieved for negative queries is up to 50% higher than negative snippets retrieved for positive or objective queries. This implies that, although the majority of search results are objective, the sentiment of a query is also reflected by the results to some extent.

5.2.6 Post-Retrieval Analysis

As mentioned before, we do not assume that an opinionated query does necessarily express the personal view of the user issuing it. For instance, a person may submit the query "abortion is a sin" to see the arguments of the people holding that opinion, or just to see whether such an opinion exists for this topic. Thus, we cannot guarantee that the existence of the query corresponds to the opinion of the user who submitted it, but we can identify that this particular opinion exists for the topic in the query and furthermore the opinion was searched for by a large number of users, as justified by the aforementioned query volume analysis. However, it is still worthwhile and illustrative to analyze the post-retrieval behavior of the user, i.e., *how* she behaves after the results are displayed. Although this cannot perfectly explain why she submitted the query, it can help to verify whether she was really looking for a particular opinion.

Setup We conducted a small-scale analysis on an MSN query log excerpt (the RFP 2006 dataset) that contains 15 million queries along with the full URL information for the clicked results. (The AOL log used in the other parts of the chapter turned out to

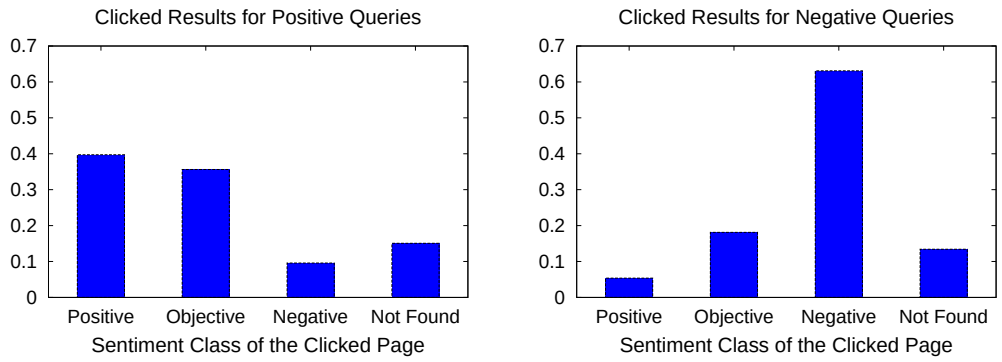


Figure 5.3: Sentiment distribution of the clicked results for (a) positive queries, and (b) negative queries. We also show the fraction of the pages that are not found, i.e., not accessible online anymore.

be useless as for the clicked results only the top-level domain of the URL is provided). We chose 5 topics ("abortion", "euthanasia", "genetic engineering", "marijuana", and "stem cell research") out of the 50 used in our paper, for which some related queries in the log could be found. For each of these topics, we annotated the queries as objective, positive or negative, yielding 79 (5%) opinionated queries among a set of 1,583 queries. For these 79 queries, there existed a total of 222 clicked URLs. For each of the clicked pages, we used the Way Back Machine[5] to get the version back from 2006 (if available) and annotated them as objective, positive or negative. In this way, we gathered and annotated 191 clicked pages.

Results Figures 5.3(a) and (b) show the percentage of clicked pages per sentiment class for the queries labeled as positive and negative. For the positive queries, the majority of the clicked pages were either positive or objective, with each class containing approx. 40% of the clicks. (The high number of objective queries is consistent with our findings above that the majority of the retrieved results by the search engines are objective, regardless of the sentiment in the query.) For the negative queries, more than 60% of the clicked pages are negative. Therefore, our results provide evidence that users who submit opinionated queries (especially negative ones) are likely to click on opinionated results in the same direction (i.e., these queries really serve as a mechanism for accessing opinionated material on the topic in question).

[5] http://web.archive.org

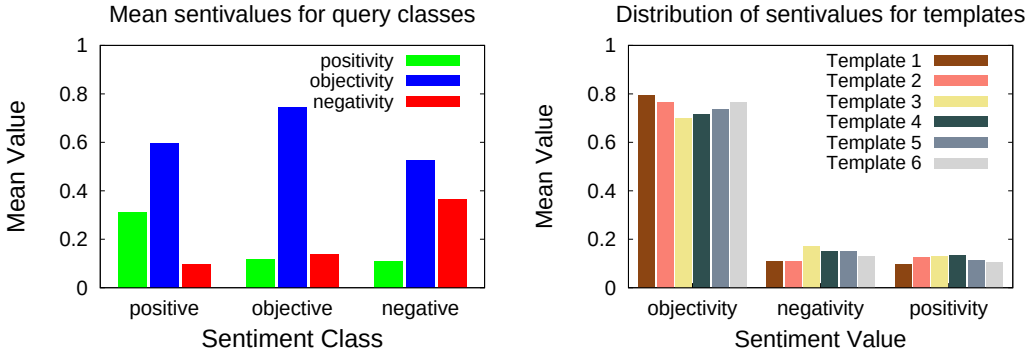


Figure 5.4: (a) Mean sentivalue scores (from SentiWordNet) in each query class, (b) Distribution of average sentivalue scores of queries (from SentiWordNet) obtained from each template.

5.2.7 Lexicon-Based Sentiment Analysis

Sentiment in Queries We also investigated whether the human judged labels for the 7,651 queries match to automatically obtained sentiment scores using the SentiWordNet thesaurus, a sentiment lexicon which was described in Section 2.2.

Using SentiWordNet we first assigned a sentivalue to each query by computing the averages of positivity, negativity and objectivity values over the adjectives extracted from the query text that have an entry in the SentiWordNet thesaurus. If an adjective appears in more than one WordNet concept (synset), the sentivalues for each occurrence are averaged to obtain the triple for this term. We used only adjectives because our experiments with additional term types (i.e., nouns and verbs) yielded less accurate results; a similar observation is also reported for the short user comments in YouTube [120]. At the end, an overall number of 2,517 queries (i.e., 31% of the manually annotated queries) was found to contain adjectives covered by SentiWordNet.

Figure 5.4(a) shows the means of positivity, negativity and objectivity scores over all queries in each class as labeled by human judges. For all three classes, we observe that the objectivity scores are rather high. This might be due to the fact that some of the positive/negative terms in the queries do not appear in the thesaurus. Nevertheless, the positively (negatively) labeled queries yield a considerably higher positivity (negativity) than negativity (positivity) score. Figure 5.4(b) shows the distribution of average sentivalues for queries from each template in Table 5.2. A comparison with Figure 5.1 reveals that automatically derived sentivalues for each template follow a similar distribution as human annotated class labels. However, the positivity and negativity scores are lower, as already discussed for Figure 5.4(a). To remedy this problem, we applied a machine learning based approach that will be discussed in Section 5.3.

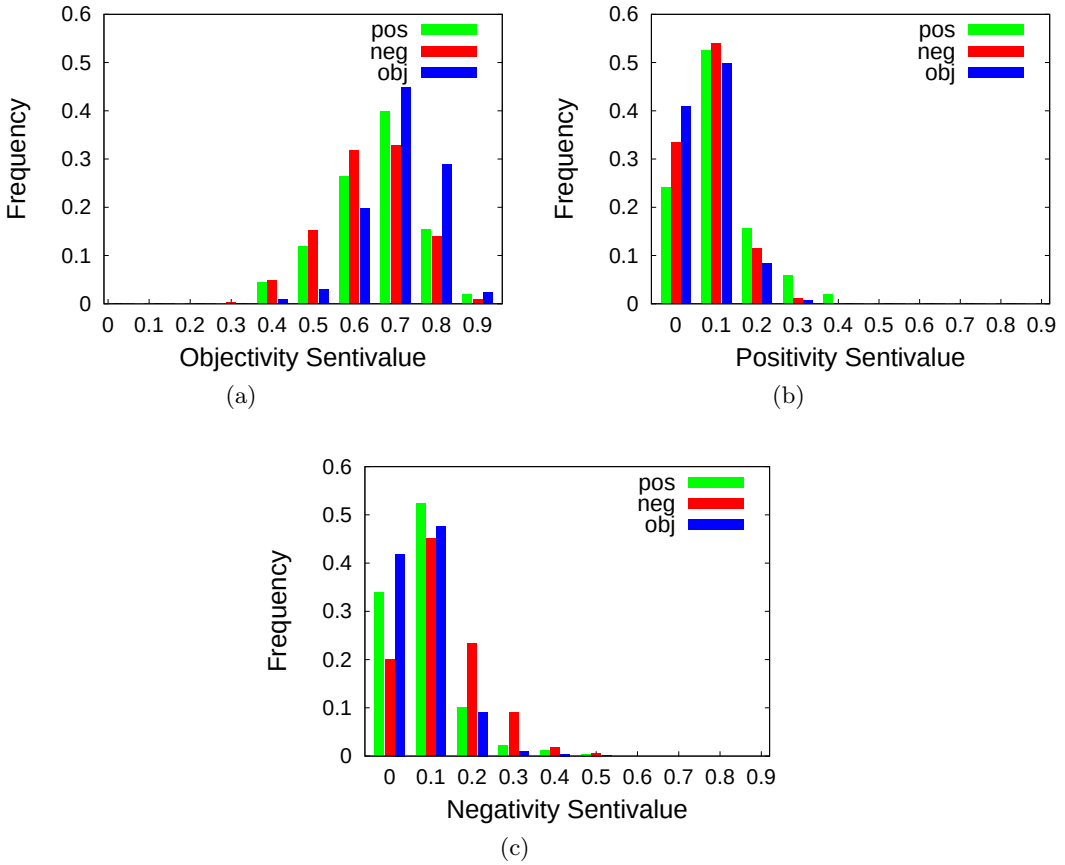


Figure 5.5: Distribution of query snippets' (a) objectivity, (b) positivity, and (c) negativity sentivalue scores (from SentiWordNet) in each query sentiment class.

Sentiment in Query Results Finally, we studied the sentiment of query results as in the previous section. In particular, for all queries in each sentiment class, we computed the sentivalues for the adjectives in the query text and the top-10 result snippets gathered via the Yahoo! API.

Figure 5.5 shows the distribution of human labeled queries from three sentiment classes across the sentivalue bins for neutrality, positivity, and negativity, respectively. The histogram in Figure 5.5(a) can be regarded as further evidence supporting the trends observed in the previous section and in [44]: the results for objective queries also yield the highest objectivity scores, especially when the score is larger than 0.7 (i.e., indicating higher confidence). In contrast, for opinionated queries, the snippets also reflect the opinion to some extent (cf. Figures 5.5(b) and 5.5(c)).

Our findings in this section reveal that even a limited-vocabulary based sentiment analysis strategy serves well in our framework and yields results quite consistent with manual annotations. In the following sections, we employ machine learning techniques for automatic sentiment analysis to facilitate the adaptation to the rapidly changing

vocabulary of Web users.

5.2.8 Regional Analysis

Opinions regarding a controversial topic can vary considerably with respect to location of the searcher and time of the search. For instance, a controversial topic such as "gay marriage" might be perceived more positively in Europe than in the Middle East. Similarly, for the same example topic, opinions have become less negative over time as social tolerance on such issues has increased. Here, we provide an analysis of the impact of region on the sentiments expressed in queries for controversial topics.

Setup We used the query collection strategy outlined in Section 5.2.1. In order to avoid issuing excessive numbers of requests to search engines, we focused on template 5 (i.e.,"⟨*topic*⟩ is [*letter*]", as shown in Table 5.2), which turned out to yield the largest fraction of opinionated queries. To obtain region-specific queries, we sent the search requests in English, German, and Spanish to the corresponding search front ends with domains extensions .com, .de and .es, respectively. For German and Spanish, we revised template 5 with the auxiliary verbs in the corresponding language. Note that the scenario of English queries submitted to the main front end of the search engine represents a rather global case, whereas queries in German or Spanish might reveal more region-specific opinions of the web users (assuming that the majority of queries in German and Spanish are submitted from the corresponding countries).

We observed that the number of opinionated queries gathered with template 5 is smaller for German and Spanish than that for English (see Table 5.5). This might be due to the unequal volumes of queries, as English queries constitute the largest query stream for most search engines. We emphasize that our choice of a fixed template for collecting opinionated queries is essentially caused by the lack of very large

Table 5.5: Topics and the number of manually annotated queries (obtained via template 5) in each of the three languages (English, German and Spanish).

Topic	**Queries**	**Topic**	**Queries**	**Topic**	**Queries**
abortion	223	abtreibung	31	aborto	95
barack obama	233	barack obama	32	barack obama	58
climate change	205	klimawandel	41	cambio climático	29
communism	208	kommunismus	27	comunismo	62
economy	227	wirtschaft	45	economía	29
homosexuality	208	homosexualität	53	homosexualidad	62
iphone	260	iphone	130	iphone	141
islam	243	islam	128	islam	52
marijuana	226	marihuana	25	marihuana	110
marriage	240	ehe	190	matrimonio	86

Table 5.6: Examples of objective, positive and negative queries in each of the three languages (English, German and Spanish) for the topic "Iphone".

Objective	Positive	Negative
	English	
iphone is xbox controller	iphone is the best smartphone	iphone is a piece of crap
iphone is made where	iphone is my life	iphone is killing the internet
iphone is eligible for upgrade	iphone is years ahead	iphone is the devil
iphone is cdma or gsm	iphone is king	iphone is killing me
iphone is made in china	iphone is better than blackberry	iphone is really slow
	German	
iphone ist im safe mode	iphone ist das beste handy	iphone ist extrem langsam
iphone ist nass geworden	iphone ist cool	iphone ist mir zu teuer
iphone ist in deutschland	iphone ist gut	iphone ist giftig
iphone ist jailbreaked	iphone ist genial	iphone ist vergangenheit
iphone ist unlocked	iphone ist zukunft	iphone ist müll
	Spanish	
iphone es celular	iphone es facil de usar	iphone es falso
iphone es gsm	iphone es el mejor celular	iphone es ilegal
iphone es el gadget	iphone es el mejor telefono	iphone es caro
iphone es un telefono	iphone es buena opcion	iphone es para gays
iphone es de 8gb	iphone es increible	iphone es muy caro

publicly available query logs. We aimed to gather queries that are more probable to be opinionated with the least possible burden to the suggestion system of the search engine.

Among our initial set of 50 topics, we identified 10 topics (listed in Table 5.5) for which all three front ends returned at least 25 queries. Next, all of the queries retrieved for these topics were manually annotated by native speakers of the corresponding language using the guidelines of Section 5.2.2. In this way we obtained 724, 702, and 2,272 annotated queries for the Spanish, German, and English front end, respectively. Table 5.6 shows a couple of example queries for each sentiment category for the topic "Iphone" (obtained via template 5) in English, German and Spanish.

Results Figures 5.6(a), (b) and (c) show the fraction of queries that are labeled as positive, negative and objective for each topic and region/language. A comparison among the figures reveals that, as expected, various topics are perceived differently among the users that submitted their queries to distinct front ends. For instance, the queries submitted in Spanish and German for the topic "climate change" exhibit a considerably higher positivity in comparison to those submitted in English to the main front-end. The higher positivity in Germany and Spain might be explained by the EU countries' leading role in developing policies related to climate change as well as the existence of supportive political groups in these countries (such as the Greens in Germany). On the other hand, the United States, from where most of the English

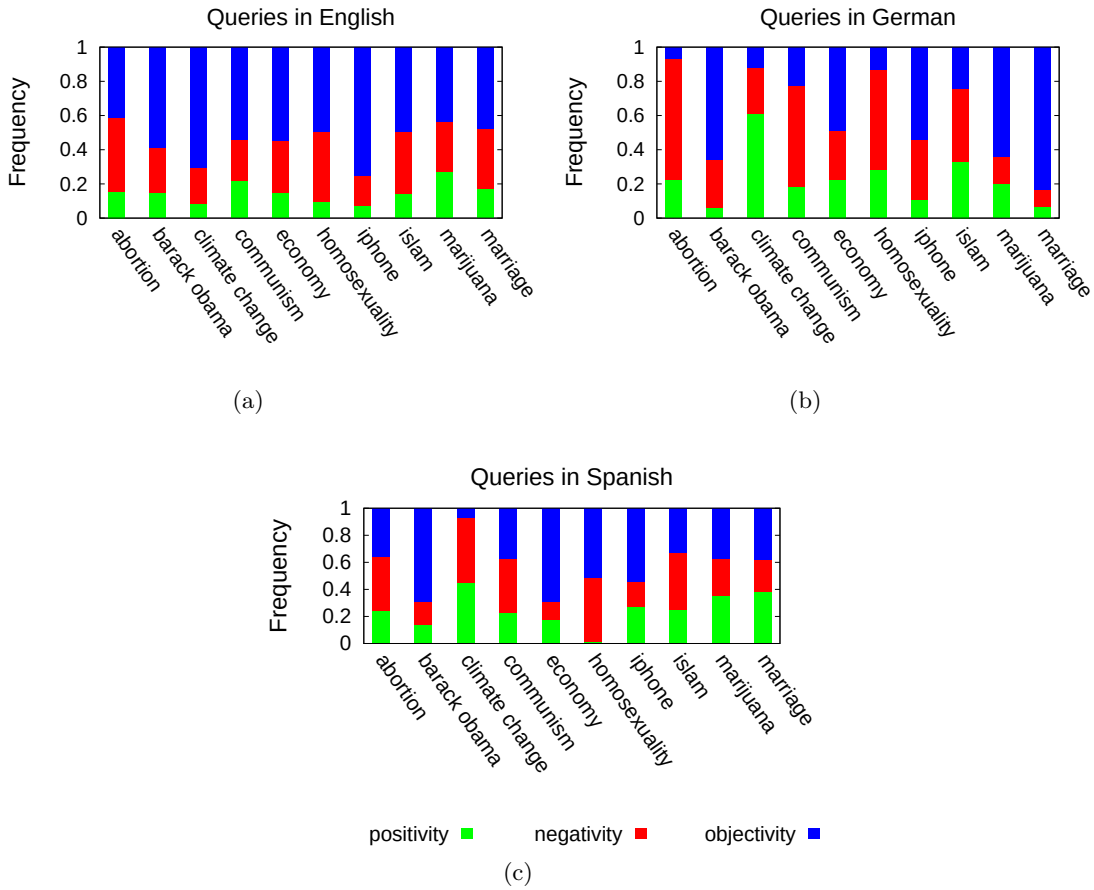


Figure 5.6: Distribution of sentiment class annotations for each topic using queries submitted in (a) English, (b) German, and (c) Spanish.

queries are possibly submitted, are known to be rather reluctant to related legislation on the issue (see e.g. [142]). While Figure 5.6 indicates noticeable differences in the perception of the topics, analyzing the underlying reasons for such differences is beyond the scope of this study and the authors' expertise. However, we believe that our findings unleash the potential of analyzing opinionated queries, which is a rather overlooked source of information up to now.

We emphasize that our findings in this section regarding the regional dynamics of opinionated queries are not comprehensive, as it is difficult to obtain datasets and ground truth from different regions and at many points in time (that is why we leave the temporal dimension as a future work). However, our examples motivate the investigation of opinionated queries and methods for automatic sentiment analysis of queries, and imply interesting applications, such as trend analysis and detection of controversial topics.

5.3 Detecting Query Sentiment

In this section, we study the application of various state-of-the-art classifiers to detect the sentiment class of a given query.

Setup For our classification experiments, we constructed feature vectors based on the top-10 result titles and snippets, in addition to query text itself. We considered 4 different representations for a given query: (i) query text only (denoted as *QText*), (ii) query text + titles for top-10 query results (*QTextTitle*), (iii) query text + snippets for top-10 query results (*QTextSnippet*), and (iv) query text + titles + snippets for top-10 query results (*QTextTitleSnippet*). We constructed multi-dimensional feature vectors using TF-IDF weights of the terms involved in each possible representation. While doing this, we also accounted for negations (i.e., if a negation, say, "not", immediately precedes another term t, we created a virtual term not_t in a similar way as described in [101]).

We used five state-of-the-art text classification approaches: simple logistic regression (SLR), multinomial Naive Bayes (mNB), SVM (SMO variant) and SVM (L2-loss linear) as implemented in the well-known Weka library [68], and the ν-Support Vector Classification (ν-SVC) formulation of SVM from LIBSVM [26]. We built three types of binary classifiers to separate each sentiment class from the other two classes, i.e. we applied a "one vs. all" (OVA) strategy. We build four different versions of each classifier based on the query representations discussed above.

For training the classifiers, we randomly split the instances from the target class into two sets reserved for training and testing, and randomly selected an equal number of instances from the remaining two classes for training as well as for test sets. In this way, we created balanced training and test sets for each classifier. This is similar to the approach employed by [13] to eliminate the effect of any underlying bias for a particular sentiment class in the data. We repeated the experiments by switching training and test sets and computed the averages for the evaluation metrics. We chose the number of training queries in such a way that the maximum number of available annotated queries could be used during the training and testing. For instance, as around 800 queries are annotated as positive, the positive vs. all classifier was trained with 400 queries from the positive class and 200 queries selected from each of the negative and objective classes. The test set was created analogously. For the negative vs. all and subjective vs. all classifiers, it was possible to use more training queries as there exist a larger number of annotated queries for these scenarios; therefore, we trained and evaluated them with 1,200 and 1,600 queries, respectively.

Results We first evaluated each classifier in terms of the classification accuracy and area under the curve (AUC) (for the Receiver Operating Characteristic (ROC) curve). The evaluation results in Tables 5.7, 5.8, and 5.9 reveal that using richer query representations (i.e., with titles and snippets) does not result in additional gains compared to simply using the query text itself. Indeed, query text alone is adequate to decide on the sentiment of a query with good accuracy, especially for the positive (negative) vs. all classifiers. This is not surprising, as users usually try to convey their information need clearly and concisely in their keyword queries. In contrast, longer texts, such as blog entries or reviews, may usually involve more sophisticated use of language (e.g., idioms, metaphors, or irony), which can make sentiment analysis more difficult [3]. A similar observation is also reported for sentiment classification in microblogs, where brevity turned out to be an advantage [13].

Table 5.7: Classification accuracy and AUC for the subjective vs. all classifiers trained with four different representations of the queries (*QAll* stands for *QTextTitleSnippet*).

	Accuracy				AUC			
	QText	QTextTitle	QTextSnippet	QAll	Text	QTextTitle	QTextSnippet	QAll
mNB	0.76	0.73	0.72	0.72	0.86	0.81	0.79	0.79
SLR	0.80	0.79	0.73	0.73	0.85	0.84	0.80	0.80
SVM (L2-LL)	0.81	0.80	0.74	0.75	0.81	0.80	0.74	0.75
SVM (SMO)	0.80	0.77	0.71	0.70	0.81	0.77	0.71	0.71
SVM (ν-SVC)	0.80	0.80	0.74	0.75	0.86	0.85	0.82	0.82

Table 5.8: Classification accuracy and AUC for the positive vs. all classifiers trained with four different representations of the queries (*QAll* stands for *QTextTitleSnippet*).

	Accuracy				AUC			
	QText	QTextTitle	QTextSnippet	QAll	QText	QTextTitle	QTextSnippet	QAll
mNB	0.68	0.63	0.62	0.61	0.75	0.68	0.66	0.65
SLR	0.71	0.66	0.62	0.62	0.76	0.70	0.65	0.66
SVM (L2-LL)	0.73	0.66	0.64	0.63	0.73	0.66	0.64	0.63
SVM (SMO)	0.72	0.68	0.61	0.61	0.72	0.68	0.62	0.61
SVM (ν-SVC)	0.73	0.70	0.64	0.64	0.81	0.76	0.70	0.71

Table 5.9: Classification accuracy and AUC for the negative vs. all classifiers trained with four different representations of the queries (*QAll* stands for *TextTitleSnippet*).

	Accuracy				AUC			
	QText	QTextTitle	QTextSnippet	QAll	QText	QTextTitle	QTextSnippet	QAll
mNB	0.72	0.67	0.66	0.66	0.80	0.73	0.71	0.71
SLR	0.73	0.67	0.63	0.62	0.79	0.72	0.66	0.67
SVM (L2-LL)	0.76	0.69	0.67	0.66	0.76	0.69	0.67	0.66
SVM (SMO)	0.77	0.69	0.63	0.63	0.77	0.69	0.63	0.63
SVM (ν-SVC)	0.76	0.71	0.64	0.67	0.84	0.78	0.70	0.73

The results in Tables 5.7, 5.8, and 5.9 show that mNB and SLR are usually inferior to SVM classifiers for the query sentiment detection task, and among the latter group of classifiers, ν-SVC (from LIBSVM) performs the best. Using only query texts, binary ν-SVC classifiers *positive vs. all*, *negative vs. all* and *subjective vs. all* yield accuracy values of 0.74, 0.76 and 0.80, respectively. Figure 5.7 shows the performance of ν-SVC classifiers for each query representation in terms of precision-recall curves and break-even points (BEPs) for these curves (i.e., precision/recall at the point where precision is equal to recall, which is also equal to F1 in that case).

The major trends are similar to previous findings; classifiers based on the query text are superior to those that make use of additional information. Result snippets seem to be slightly useful for distinguishing subjective queries from the objective ones at low recall values (i.e., up to 0.40). Actually, scenarios that allow trading recall against precision are perfectly supported by all classifiers; for instance, for the positive vs. all and negative vs. all classifiers based on the query text, precision values remain over 0.9 up to a recall level of 0.4. This can be useful for finding specifically strong candidates

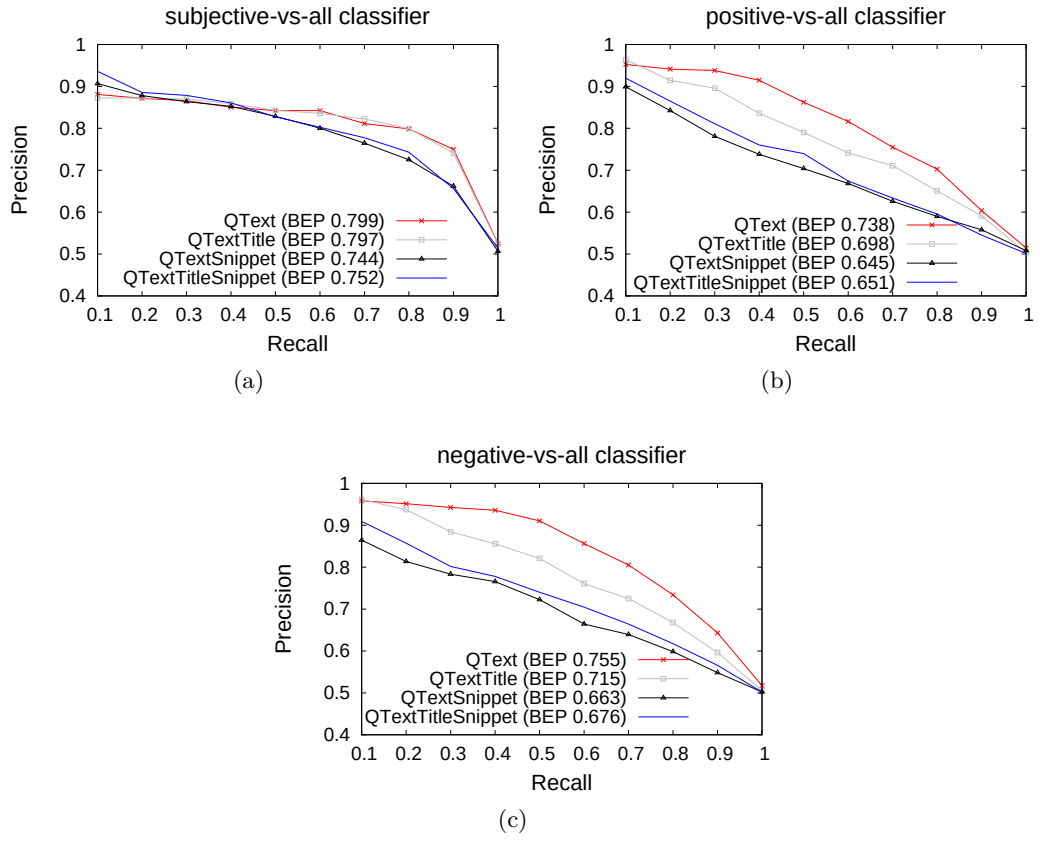


Figure 5.7: Precision-recall curves and BEPs for (a) subjective vs. all, (b) positive vs. all, and (c) negative vs. all classifiers.

of opinionated queries in large query logs.

For the best performing query representation, namely *QText*, we also applied two well-known lexicon based methods from the literature: SentiWordNet (SWN) [53] and SentiStrength [127]. We used the test queries employed in each of the classification tasks above. SWN yields accuracy values of 0.65, 0.63, and 0.65 for the test sets employed in positive-, negative- and subjective-vs-all classification experiments, respectively. SentiStrength yields slightly lower accuracy (0.62) than SWN, for positive vs. all, but is superior to the latter method at detecting negative and subjective queries, resulting in accuracy values of 0.72 and 0.68, respectively. Nevertheless, these figures are considerably lower than those for the machine learning based strategies (a finding that confirms Figure 1 in [13]); therefore, we do not discuss the lexicon based methods for query sentiment detection in the rest of this section.

Do classification models generalize to new topics?

Setup We repeated the entire experimental procedure by applying a topic-wise split of training and test sets. To this end, queries from the first 25 topics were used for training and those from the second half were used for testing (again, by selecting equal number of queries and also taking into account the number of annotated queries from each sentiment class) and vice versa. We employed the best-performing classifier in the above experiments, namely, ν-SVC.

Results The trends observed are similar as for our previous classification experiments. We obtain prec=0.89 for recall=0.2, and prec=0.83 for recall=0.4 for the positive vs. all classification task using query terms (i.e., *QText*), indicating that it is possible to trade recall against precision for better applicability. (We discarded PR-curves for these experiments for brevity.) Our findings show that even for previously unseen topics, our classifiers perform well at detecting the sentiment in queries. Furthermore, even if new contexts that require annotating additional queries may arise in time, annotating the sentiment in queries would be a less labor-intensive task than annotating full-length documents. This is another reason for exploiting sentiment in queries as proposed in this chapter.

5.4 Application Scenarios

5.4.1 Query Recommendation

There are various interesting scenarios that can benefit from detection of sentiment in Web queries. In this section we focus on the task of recommending additional

queries for opinionated queries as a use case. More specifically, we investigate the potential of improving the relevance of suggested queries by analyzing the sentiment in the submitted query, and suggesting queries in the same direction.

Recommender Methods For the query recommendation scenario, we first trained a positive (negative) vs. all sentiment classifier in a leave-one-topic-out manner, i.e., by using 49 topics for training and one for testing. We used balanced sets with equal number of randomly selected instances from each class. Next, we ranked queries for each topic based on the distance from the separating SVM hyperplane, and used the query classified as positive (negative) with the highest confidence as seed query for the topic. Then, we generated query suggestions for each seed query in two ways: 1) As a baseline, we issued the seed query to a major search engine in October 2011 (i.e., the same one used in the other parts of this chapter), and collected all suggestions that were obtained through auto-completions, or recommended on the result page under "related queries" (only the top-10 were selected). We name this set "search engine suggestions". 2) We selected the same number of queries from the distance ranked list of classified queries for the same topic (except for the seed query itself). We refer to this set as "opinionated suggestions".

Setup For evaluating the query recommendations, we shuffled both suggestion sets and conducted a user study where subjects were asked to label each query as relevant/irrelevant/undecided with respect to the seed query. To reduce the manual workload of the participants, we decided to consider only 15 out of the 50 topics with highest polarity with respect to the ground truth annotations discussed in Section 5.2.2. We gathered the manual assessments using two sets of annotators: in-house annotators and annotators from a crowdsourcing platform.

For the in-house annotations, five computer science researchers/students were involved who were not aware of the final goal of our evaluation. Each of the human judges was randomly assigned 6 seed queries along with the suggestions. We made sure that the seed queries from the same topic were assigned to different judges. On average, the participants annotated a combined list of around 20 suggestions per seed query and topic. We considered two seed queries (i.e., the most positive and negative ones) for each of the 15 topics, yielding 600 annotated suggestions for 30 seed queries.

For crowdsourced annotations, we created a Human Intelligence Task (HIT) at Amazon Mechanical Turk (AMT) for each pair of a seed query and a suggested query, where we asked the workers to label the suggestion as relevant, irrelevant or undecided. Each HIT was assigned to five different workers and the final decision was computed based on majority voting.

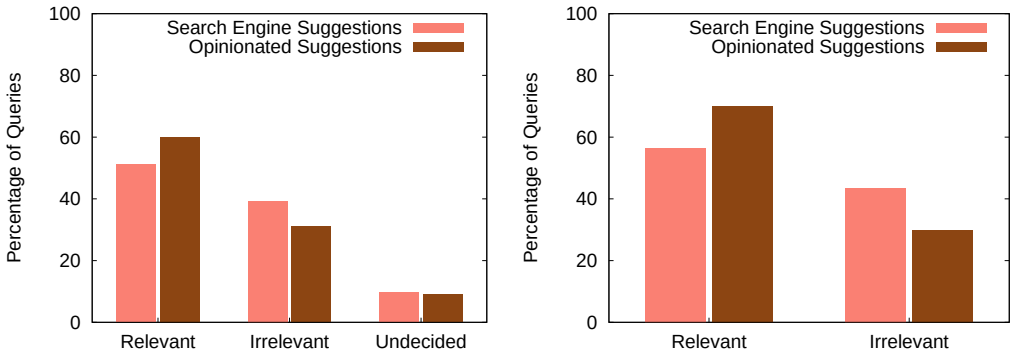


Figure 5.8: Query recommendation performance based on (a) in-house annotations, and (b) AMT annotations.

Results Figures 5.8(a) and (b) show the average percentage of "search engine suggestions" and "opinionated suggestions" that are labeled as relevant, irrelevant or undecided by the in-house and AMT annotators, respectively. In contrast to the in-house study where each suggestion is labeled by one annotator, the AMT evaluation provides five annotations per suggestion and hence, no query suggestion is labeled as "undecided" after the majority voting (see Figure 5.8(b)). Still, 67.3% of the AMT annotations agree with those of the in-house annotations. The Fleiss's Kappa for the inter-user agreement among the AMT workers was found to be 0.68. Figures 5.8(a) and (b) reveal that the opinionated suggestions are more relevant than the original recommendations from the search system. We applied an unpaired t-test to compare whether the difference between the mean relevance scores (based on the in-house annotations) of two recommendation lists is significant (assuming that all undecided queries are also irrelevant), and found that our improvements are statistically significant on a 95% confidence level (with df=628, $|t|$=2.01). We verified the statistical significance also for the results based on the annotations of AMT workers (with df=595, $|t|$=3.56). Furthermore, we observed that there is only little overlap between the two recommendation lists, indicating that result set merging can further improve the relevance of query recommendations.

A detailed inspection of the results provided further interesting insights. We noticed that for most of the seed queries, there are no auto-completions provided, which are usually very accurate; instead, only a list of, more error prone, "related queries" is shown. In practice, our sentiment based recommendation mechanism can be applied in situations where no or few auto-completions are available (see Table 5.10). We also observed that the top-10 search results vary largely for the different recommended queries listed in Table 5.10, confirming that this type of query reformulation can provide

Table 5.10: Search engine's suggestions (provided as "related queries" and "auto-completions", the latter are shown in italics) vs. opinionated suggestions for the query "economy is really bad".

Search Engine Suggestions	**Opinionated Suggestions**
economy is really bads	economy is bad
economy is really bad right now	why is economy bad
economy is really bad 2009	economy is still bad
economy is really band	economy is very bad now
economy is really good	economy is getting worse
economy is really funny	economy is obama's fault
economy is really bag	economy is worse than divorce
economy is really dirty	economy is killing people
gdp is really bad	economy is destroyed
mileage is really bad	economy is going to get worse

additional information and perspectives. While our primary goal here is to provide more relevant recommendations that are aligned with the sentiment of the seed query, our approach can also be employed for improving the diversity of recommendations (as in [124]) by suggesting queries in the directions other than the user's opinion. We plan to explore the combination of our sentiment-based approach with original search engine suggestions and other query recommendation approaches/scenarios in our future work.

5.4.2 Controversial Topic Discovery

Trend analysis on opinionated digital data is an emerging area that has drawn substantial attention over the last years [64, 5]. We anticipate that controversial topic discovery can be an important stage of trend analysis studies, as it sheds light on the issues on which Web users have diverse opinions. For instance, the drift of controversial topics discussed in a society over time may indicate underlying changes in its value system.

Controversial topics can have regional and temporal aspects. Mining opinionated text on the Web such as blogs or reviews in order to discover controversial topics for a particular region of the world and for a specific time period would require sophisticated and expensive techniques to accurately detect these spatio-temporal features as well as the sentiment expressed in relatively long and complicated articles. First, one has to deal with the traditional and hard problem of capturing the sentiment from a natural language text (e.g. a blog post discussing a popular product such as the iPhone) with metaphors and ironical statements. Second, there is the issue of capturing the time and region the post is intended for (for instance, the post might be discussing an older or future version of the product, or identifying problems specific for a particular country which could even differ from the host country where the blog is published).

We envision that the huge volume of queries submitted to Web search engines can be

employed for opinion mining with considerably less effort. It is easy to associate queries with a particular region, as search engines already keep track of the search front ends to which a query is submitted for localization and personalization purposes. Furthermore, queries can often be seen as short and concise statements about the topics in question. Therefore, query logs accumulated over a sufficiently long period from different search front ends can serve as an invaluable resource for discovering controversial topics in a desired region and time period. In this section, we show the applicability of our query sentiment classifiers in this context.

Topic Discovery Method In an idealistic setup, it would be sufficient to classify the sentiment of each query in the query logs of a search engine to infer potentially controversial topics. Since such large-scale query logs are not publicly available we devised a two-stage selection and filtering strategy (cf. Figure 5.9) to provide a proof-of-concept for this scenario, instead. In the first stage, the *candidate topic generation* step, we formed a set of queries that were prefixed by any combination of three letters from the English alphabet and followed by the term "is" (e.g., "mil is"). Next, we trained a machine learning model as discussed in Section 5.3 to distinguish queries with positive and negative sentiments (we dismissed the objective class as it is not useful for our purposes in this section). The queries in this set were sorted with respect to the classifier scores, from the most positive instances to the most negative ones. Finally, we selected the queries from the top and bottom $P\%$ of the list formed in the previous step, removed the part starting with "is", and grouped with respect to the remaining topic names. Those topic names that appeared more than K times in our set were identified as *candidate topics*.

We observed that while a too high K yields only few topics, a too low value of K produces rather noisy topics. For instance, for the template "lef is", suggestions include "**lef**t is right" and "**lef**t is seldom right", which clearly state an opinion about the political left. However, also the query "**lef**t ventricular hypertrophy is reversible" is suggested, which might be classified as rather positive, but probably does not refer to a controversial topic. Therefore, in the second (*controversy discovery*) stage, we collected all query suggestions using template 5 (i.e., <topic> is [letter], as shown in Table 5.2) for the candidate topics and again classified them using our classifier, to verify whether a large number of opinionated queries existed for the given topic. As might be expected, for the above example, the candidate topic "left" yields a large number of query suggestions that are opinionated, whereas "left ventricular hypertrophy" yields none. In this step, we chose topics that had at least N opinionated queries, along with the classifier scores for these queries. Finally, we ranked the topics according to the

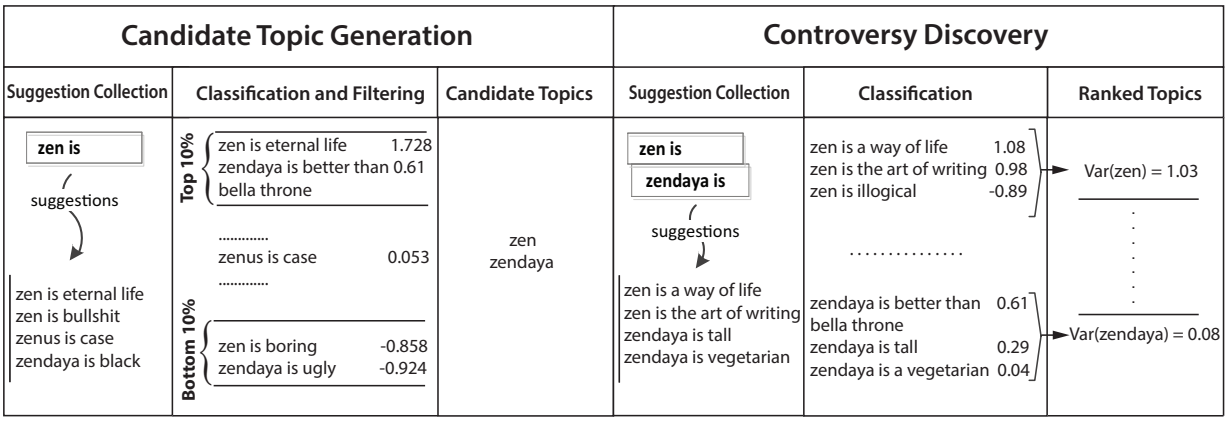


Figure 5.9: A toy example illustrating controversial topic detection: the procedure will output only "zen" as being controversial, as it yields very high variance in query sentiment scores and filter "zendaya", as its queries have less variance.

variance of the classifier scores, envisioning that topics with a higher variance in query sentiments would be more controversial. The entire process is illustrated in Figure 5.9.

In this work, we set parameter P to 10%, K to 2, and N to 50, in an ad hoc manner. The initial query suggestion set includes 98,359 queries, and almost one third of them could be classified by our detector (for the rest, none of the terms apart from the topic name appeared in the trained model). After applying these steps, we ended up with a ranked list of 273 topics from which we also removed stopwords and adjectives, resulting in an overall number of 263 topics. Note that although adjectives are very important in the sentiment detection step they usually do not correspond to actual topic *labels*. The variance scores in this list starts from 1.0, representing a probability of high-controversy and drops to 0.3 at the end of the list, corresponding to a potentially non-controversial topic.

Results The top-20 (controversial) and bottom-20 (non-controversial) topics are shown in Table 5.11. The most controversial topic, *dairy*, reflects a popular and hot debate on whether food products produced from the milk of mammals are healthy or not. *Wicca* is defined as a modern pagan religion in Wikipedia that gives rise to contradicting opinions, as some people apply attributes like *fake*, *evil*, or *stupid*, whereas others think that it is *cult*, *good* and *right*. *Splenda* is an artificial sweetener and *msg* is short for monosodium glutamate used as a flavor enhancer in food, both of which seem to trigger highly polarized views. Notice that, except for the topic *left*, which has been a controversial issue in politics for centuries, the other 4 topics among the top-5 discussed above can not be easily detected, and reflect the Web searchers' popular discussion issues at the time of this experiment. In this sense, we believe that our methodology serves well to discover topics otherwise unrevealed that cause controversy within the

Table 5.11: Topics ranked with respect to the variance in sentiment scores of their queries.

Top-20 topics		**Bottom-20 topics**	
dairy	ritalin	wood	sitting
wicca	lie	ignorance	yesterday
left	oatmeal	jesus	egg
splenda	vanity	icp	danger
msg	lsd	insanity	justice
hunting	acid	ncis	hell
euthanasia	lying	africa	weird
losing	skateboarding	registry	pakistan
lust	liz	beauty	all i do
abortion	living	pope	truth

Web community. In contrast, the bottom-20 topics seem to be less-controversial, as topics in this group are more likely to attract either mostly negative or mostly positive attitude (if they ever cause any polarity). For instance, *wood* is mostly viewed neutrally whereas *ignorance* is mostly perceived negatively. Similarly, *ICP*, a rap band in the US, seems to have a mostly negative reputation.

Quantitative Evaluation Setup For a systematic evaluation of our strategy for controversial topic discovery, we selected the top- and bottom-50 topics and conducted a user study, where each annotator was given a shuffled list of the resulting 100 topics and asked to label the topics as either controversial (denoted with 1) or non-controversial (denoted with 0). We examined if the top-50 topics were significantly more controversial than those in the bottom-50. In order to avoid personal bias, we emphasized that the annotators should not rely on their own perception of a topic, but rather decide on the possibility of existence of large groups of people that would have opposing views on the topic. As in the previous sections, we employed both in-house annotators (4 computer science researchers) and AMT workers (5 Turkers).

Quantitative Evaluation Results For each topic, we decided on the final label (controversial or not) using majority voting, and then computed the grand averages for the top-50 and bottom-50 topics. For the in-house annotations, we found the scores of 0.62 and 0.36 for the top-50 and bottom-50 topics, respectively. Similarly, AMT annotations yielded a score of 0.58 (0.32) for the top-50 (bottom-50) topics. An unpaired t-test showed the statistical significance of the difference of the population mean values on a 95% confidence level for both sets of annotations, with df = 98 and $|t|=2.67$ for the in-house participants and with df = 98 and $|t|=2.68$ for the AMT workers. We observed Fleiss' kappa coefficients [66] of 0.34 and 0.42 for inter-user

agreement among the in-house annotators and AMT workers, respectively. (Note that according to Fleiss' definition, $\kappa < 0$ corresponds to no agreement, $\kappa = 0$ to agreement by chance, and $0 < \kappa \leq 1$ to agreement beyond chance.) Also note that, 80% of the labels assigned by AMT workers overlap with those assigned by the in-house annotators. The result of this study provides further evidence for the potential of our controversial topic discovery strategy in a real-life setup.

5.5 Summary and Contributions

Our key contributions in this chapter are as follows.

We are the first to provide a detailed *analysis of sentiment in Web queries* on controversial topics. To this end, we employ a number of different query templates on the query suggestion service of a major search engine as well as a publicly available query log to obtain a large and representative sample of real user queries. Using this dataset, we conduct manual and lexicon-based analyses of sentiments in the queries, and provide answers to various research questions: To what extent can Web queries include opinions (this may or may not reflect the query issuer's own opinion)? To what extent is sentiment in the queries mirrored in retrieved results and user clicks? Is sentiment in the queries correlated with the geographical locations of users?

Secondly, we study the applicability of state-of-the-art sentiment analysis methods (including both lexicon-based and machine learning based methods) for *detecting the sentiment of the queries*. Query texts exhibit inherently different characteristics in comparison to classical corpora used for sentiment analysis (i.e., news stories, blogs, product reviews, comments, and even tweets). In this chapter, we use features obtained from the top-ranked result titles and snippets, as well as the pure query text, while applying and evaluating the current sentiment detection techniques for this new source of data with its unique characteristics. The performance is evaluated on more than 7,651 human annotated queries for 50 controversial topics.

As a final contribution, we employ our query sentiment detectors in two of the scenarios discussed above, namely, *query recommendation* and *controversial topic discovery* (for trend analysis). In extensive user studies including both in-house participants and workers from a crowdsourcing platform we show the viability of sentiment detection for both applications.

To sum up, we believe that our study accomplishes its objectiv of identifying Web search queries as a new and rich source of community information for detecting and exploiting sentiments.

6 Conclusions and Future Work

The studies presented in this thesis focused on mining, understanding and exploiting various types of community feedback in online communities and query logs. In this final section we conclude our major findings and point the reader to possible future work.

In **Chapter 3** we conducted an in-depth comment analysis to shed light on different aspects of comment ratings for YouTube and Yahoo! News platforms. Our large-scale studies using more than 11 million comments revealed interesting dependencies between the different user sentiments expressed in comments and the comment ratings. We further detailed our analysis of comment ratings and studied controversial comments that attract a comparable number of likes and dislikes from the community. In our classification experiments, we demonstrated that community feedback in social sharing systems in combination with term features in comments can be used for automatically determining the comment-centric community acceptance. Interestingly, the textual content of a comment can also be used to predict the comments that start a discussion thread.

The work in Chapter 3 further showed how the variance of ratings for a specific piece of content serves as a strong social signal to pinpoint polarizing content. Finally, we trained machine-learning models able to effectively detect trolls based on the commenting history of users. Note that, detecting troll users is an important step towards improving experience and increasing user engagement within the online communities.

Future work We believe that the findings from this chapter can provide important design guidelines for building more engaging and usable online communities. Consider the vast amount of comments attracted by popular content; our results show that we can infer the potential acceptance of comments, making it possible to enhance relevant comment discovery by highlighting the ones that are likely to be the most liked, disliked, or both (in the case of controversial comments) as well as the most replied, which are likely to trigger interesting discussions. These results might be used to provide

multiple facets (e.g. predicted controversy or community acceptance) to complement pure textual queries. In addition, comment search engines could leverage our results for deriving ranking schemes that promote comments based on their ability to attract replies and votes, as a way to give visibility. We think that integration and user evaluation within a wider system context and encompassing additional complementary retrieval and mining methods is of high practical importance.

In **Chapter 4** we provide the first comprehensive investigation for the impact of social features on video retrieval effectiveness. To this end, we focus on a keyword-based video search scenario for YouTube, and treat the popular (i.e., head and/or torso) and tail queries submitted to YouTube separately in our analyses and experiments. The social features employed in this work are derived from the raw metadata fields of the videos as well as the profiles of users who share and/or interact with these items. We show that while the basic features relying on the similarity of the query to the video titles, tags and descriptions are the most effective for the video retrieval, the social features are also valuable and can yield the best rankings for up to 58% of the queries, indicating their potential to improve the retrieval effectiveness. Our evaluations using two greedy feature selection methods and six state-of-the-art LETOR algorithms support our hypothesis: the rankers based on the subsets of features including both basic and social features outperform those built by using only the basic features. Furthermore, the social features are shown to be useful not only for the popular queries, for which there might be additional clues obtained from the abundant user click data, but also for the tail queries, for which such click information is very scarce. This latter finding is important, given that the competition among the search engines is becoming more focused on queries in the long tail (e.g., [148]).

Future work We have several future work directions for the exploration of social signals in online communities. First, we plan to extend our work to include the datasets obtained from other content sharing platforms and explore the generalizability of our findings. Second, we aim to obtain larger annotated datasets by leveraging the popular crowd-sourcing solutions. Finally, we plan to develop new LETOR strategies that are specialized for different feature types.

In **Chapter 5** we conducted an in-depth analysis to clear up different aspects of community sentiments in Web queries. Our work focused on publicly available query log as well as query suggestion data, gathered from a major search engine. We investigated how frequent are opinions and sentiments expressed in queries and how query sentiment can depend on the regional context. We trained models for detecting with high precision the sentiment of queries. In our classification experiments, we demon-

strated that using query text alone is often sufficient for automatically determining the sentiment of queries. This makes a large-scale sentiment-oriented analysis of query logs feasible, and opens many avenues for opinion mining in query logs. Finally, we showed that automatic sentiment analysis of queries can be applied for discovering controversial topics and for recommending related queries to the user.

Future work Directions of our future work involve investigating the benefits of query sentiment detection in other scenarios such as result aggregation, trend analysis, and targeted advertising. We will also focus on studying the temporal development of opinions based on the sentiment in the queries issued by the search engine users.

Bibliography

[1] Aggarwal, C., and Zhai, C. A survey of text classification algorithms. In *Mining Text Data*, C. C. Aggarwal and C. Zhai, Eds. Springer US, 2012, pp. 163–222.

[2] Agichtein, E., Castillo, C., Donato, D., Gionis, A., and Mishne, G. Finding high-quality content in social media. In *Proceedings of the 2008 International Conference on Web Search and Data Mining* (2008), WSDM '08, ACM, pp. 183–194.

[3] Ahmad, K. *Affective Computing and Sentiment Analysis: Emotion, Metaphor and Terminology (Text, Speech and Language Technology)*, 1st edition. ed. Springer, Aug. 2011.

[4] Alcântara, O. D. A., Jr., Á. R. P., de Almeida, H. M., Gonçalves, M. A., Middleton, C., and Baeza-Yates, R. A. Wcl2r: A benchmark collection for learning to rank research with clickthrough data. *JIDM 1*, 3 (2010), 551–566.

[5] Allan, J. *Topic Detection and Tracking: Event-Based Information Organization*. Kluwer Academic Publishers, 2002.

[6] Alonzo, M., and Aiken, M. Flaming in electronic communication. *Decis. Support Syst. 36*, 3 (Jan. 2004), 205–213.

[7] Anagnostopoulos, A., Becchetti, L., Castillo, C., and Gionis, A. An optimization framework for query recommendation. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining* (2010), WSDM '10, ACM, pp. 161–170.

[8] Awadallah, R., Ramanath, M., and Weikum, G. Harmony and dissonance: organizing the people's voices on political controversies. In *Proceedings of the fifth ACM international conference on Web search and data mining* (2012), WSDM '12, ACM, pp. 523–532.

[9] Baeza-Yates, R., Hurtado, C., and Mendoza, M. Query recommendation using query logs in search engines. In *Proceedings of the 2004 International*

Conference on Current Trends in Database Technology (2004), Springer-Verlag, pp. 588–596.

[10] Baeza-Yates, R. A., and Ribeiro-Neto, B. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[11] Bar-Yossef, Z., and Gurevich, M. Mining search engine query logs via suggestion sampling. *Proc. VLDB Endow. 1* (Aug. 2008), 54–65.

[12] Bar-Yossef, Z., and Kraus, N. Context-sensitive query auto-completion. In *Proceedings of the 20th International Conference on World Wide Web* (2011), WWW '11, ACM, pp. 107–116.

[13] Bermingham, A., and Smeaton, A. F. Classifying sentiment in microblogs: is brevity an advantage? In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (2010), CIKM '10, ACM, pp. 1833–1836.

[14] Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[15] Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *J. Mach. Learn. Res. 3* (Mar. 2003), 993–1022.

[16] Broder, A. A taxonomy of web search. *SIGIR Forum 36*, 2 (Sept. 2002), 3–10.

[17] Broder, A. Z., Carmel, D., Herscovici, M., Soffer, A., and Zien, J. Efficient query evaluation using a two-level retrieval process. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management* (New York, NY, USA, 2003), CIKM '03, ACM, pp. 426–434.

[18] Broder, A. Z., Fontoura, M., Gabrilovich, E., Joshi, A., Josifovski, V., and Zhang, T. Robust classification of rare queries using web knowledge. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2007), ACM, pp. 231–238.

[19] Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., and Hullender, G. Learning to rank using gradient descent. In *Proceedings of the 22Nd International Conference on Machine Learning* (New York, NY, USA, 2005), ICML '05, ACM, pp. 89–96.

[20] Cambazoglu, B. B., Zaragoza, H., Chapelle, O., Chen, J., Liao, C., Zheng, Z., and Degenhardt, J. Early exit optimizations for additive machine learned ranking systems. In *WSDM* (2010), ACM, pp. 411–420.

[21] Cao, H., Hu, D. H., Shen, D., Jiang, D., Sun, J.-T., Chen, E., and Yang, Q. Context-aware query classification. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2009), ACM, pp. 3–10.

[22] Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning* (2007), ICML '07, ACM, pp. 129–136.

[23] Carbonell, J., and Goldstein, J. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (1998), SIGIR '98, ACM, pp. 335–336.

[24] Cha, M., Kwak, H., Rodriguez, P., Ahn, Y.-Y., and Moon, S. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC '07, ACM, pp. 1–14.

[25] Cha, M., Kwak, H., Rodriguez, P., Ahn, Y.-Y., and Moon, S. Analyzing the video popularity characteristics of large-scale user generated content systems. *IEEE/ACM Trans. Netw. 17*, 5 (2009), 1357–1370.

[26] Chang, C.-C., and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology 2* (2011), 27:1–27:27. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[27] Chapelle, O., and Chang, Y. Yahoo! learning to rank challenge overview. *Journal of Machine Learning Research - Proceedings Track 14* (2011), 1–24.

[28] Chapelle, O., Metlzer, D., Zhang, Y., and Grinspan, P. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management* (New York, NY, USA, 2009), CIKM '09, ACM, pp. 621–630.

[29] Chelaru, S., Altingovde, I. S., and Siersdorfer, S. Analyzing the polarity of opinionated queries. In *Proceedings of the 34th European Conference on IR Research* (2012), ECIR '12, Springer-Verlag, pp. 463–467.

[30] Chelaru, S., Altingovde, I. S., Siersdorfer, S., and Nejdl, W. Analyzing, detecting, and exploiting sentiment in web queries. *ACM Transactions on the Web 8*, 1 (Dec. 2013), 6:1–6:28.

[31] Chelaru, S., Herder, E., Djafari Naini, K., and Siehndel, P. Recognizing skill networks and their specific communication and connection practices. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media* (2014), HT '14, pp. 13–23.

[32] Chelaru, S., Orellana-Rodriguez, C., and Altingovde, I. How useful is social feedback for learning to rank youtube videos? *World Wide Web Journal* (2013), 1–29.

[33] Chelaru, S., Orellana-Rodriguez, C., and Altingovde, I. S. Can social features help learning to rank youtube videos? In *Proceedings of the 13th International Conference on Web Information Systems Engineering* (2012), WISE '12, Springer-Verlag, pp. 552–566.

[34] Chelaru, S., Stewart, A., and Siersdorfer, S. Exploiting an inferred comment graph for clustering videos in youtube. In *GLocal Report* (2011).

[35] Cheng, X., Dale, C., and Liu, J. Understanding the characteristics of internet short video sharing: Youtube as a case study. In *Technical Report arXiv:0707.3670v1 cs.NI* (New York, NY, USA, 2007), Cornell University, arXiv e-prints.

[36] Cheng, X., Dale, C., and Liu, J. Statistics and social network of youtube videos. In *Proc. of IEEE IWQoS'08* (2008).

[37] Choudhury, M., Sundaram, H., John, A., and Seligmann, D. What makes conversations interesting? themes, participants and consequences of conversations in online social media. In *Proceedings of the 18th international conference on World Wide Web* (2009), WWW '10, ACM, pp. 331–340.

[38] Cortes, C., and Vapnik, V. Support-vector networks. *Mach. Learn. 20*, 3 (Sept. 1995), 273–297.

[39] Cunningham, S. J., and Nichols, D. M. How people find videos. In *Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries* (2008), JCDL '08, ACM, pp. 201–210.

[40] Dalal, O., Sengemedu, S. H., and Sanyal, S. Multi-objective ranking of comments on web. In *Proceedings of the 21st International Conference on World Wide Web* (2012), WWW '12, ACM, pp. 419–428.

[41] Danescu-Niculescu-Mizil, C., Kossinets, G., Kleinberg, J., and Lee, L. How opinions are received by online communities: A case study on amazon.com helpfulness votes. In *Proceedings of the 18th International Conference on World Wide Web* (2009), WWW '09, ACM, pp. 141–150.

[42] Dang, V., and Croft, W. B. Feature selection for document ranking using best first search and coordinate ascent. In *Proc. of SIGIR'10 Workshop on Feature Generation and Selection for Information Retrieval* (2010).

[43] Davidson, J., Liebald, B., Liu, J., Nandy, P., Van Vleet, T., Gargi, U., Gupta, S., He, Y., Lambert, M., Livingston, B., and Sampath, D. The youtube video recommendation system. In *Proceedings of the Fourth ACM Conference on Recommender Systems* (2010), RecSys '10, ACM, pp. 293–296.

[44] Demartini, G., and Siersdorfer, S. Dear search engine: what's your opinion about...?: sentiment analysis for semantic enrichment of web search results. In *Proceedings of the 3rd International Semantic Search Workshop* (2010), ACM, pp. 4:1–4:7.

[45] Demartini, G., Siersdorfer, S., Chelaru, S., and Nejdl, W. Analyzing political trends in the blogosphere. In *Proceedings of the Fifth International Conference on Weblogs and Social Media* (2011), ICWSM '11.

[46] Demartini, G., Siersdorfer, S., Chelaru, S., and Nejdl, W. Exploiting the blogosphere to estimate public opinion in the political domain. In *GLocal Report* (2011).

[47] Denecke, K. Using sentiwordnet for multilingual sentiment analysis. In *ICDE Workshops* (2008), pp. 507–512.

[48] Drucker, H., Wu, D., and Vapnik, V. N. Support vector machines for spam categorization. *IEEE TRANSACTIONS ON NEURAL NETWORKS 10*, 5 (1999), 1048–1054.

[49] Dumais, S., Platt, J., Heckerman, D., and Sahami, M. Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management* (1998), CIKM '98, ACM, pp. 148–155.

[50] Dumais, S., Platt, J., Heckerman, D., and Sahami, M. Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management* (1998), CIKM '98, ACM, pp. 148–155.

[51] Easley, D., and Kleinberg, J. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA, 2010.

[52] Esuli, A. Automatic generation of lexical resources for opinion mining: models, algorithms and applications. *SIGIR Forum 42* (November 2008), 105–106.

[53] Esuli, A., and Sebastiani, F. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation* (2006), LREC '06, pp. 417–422.

[54] Fagin, R., Kumar, R., and Sivakumar, D. Comparing top k lists. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (2003), SODA '03, Society for Industrial and Applied Mathematics, pp. 28–36.

[55] Fellbaum, C., Ed. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[56] Filippova, K., and Hall, K. B. Improved video categorization from text metadata and user comments. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2011), SIGIR '11, ACM, pp. 835–842.

[57] Fonseca, B. M., Golgher, P. B., de Moura, E. S., and Ziviani, N. Using association rules to discover search engines related queries. In *Proceedings of the First Conference on Latin American Web Congress* (2003), LA-Web '03, IEEE Computer Society, pp. 66–71.

[58] Freund, Y., Iyer, R., Schapire, R. E., and Singer, Y. An efficient boosting algorithm for combining preferences. *J. Mach. Learn. Res. 4* (2003), 933–969.

[59] Friedman, J. H. Stochastic gradient boosting. *Comput. Stat. Data Anal. 38*, 4 (Feb. 2002), 367–378.

[60] Geng, X., Liu, T.-Y., Qin, T., and Li, H. Feature selection for ranking. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2007), SIGIR '07, ACM, pp. 407–414.

[61] Giannopoulos, G., Weber, I., Jaimes, A., and Sellis, T. K. Diversifying user comments on news articles. In *Proceedings of the 13th International Conference on Web Information Systems Engineering* (2012), WISE '12, Springer-Verlag, pp. 100–113.

[62] Gill, P., Arlitt, M., Li, Z., and Mahanti, A. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), IMC '07, ACM, pp. 15–28.

[63] Gómez, V., Kaltenbrunner, A., and López, V. Statistical analysis of the social network and discussion threads in slashdot. In *Proceedings of the 17th international conference on World Wide Web* (2008), WWW '08, ACM, pp. 645–654.

[64] Goorha, S., and Ungar, L. Discovery of significant emerging trends. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2010), KDD '10, ACM, pp. 57–64.

[65] Grace, J., Gruhl, D., Haas, K., Nagarajan, M., Robson, C., and Sahoo, N. Artist ranking through analysis of online community comments. Tech. rep., IBM Research Technical Report, 2008.

[66] Gwet, K. *Handbook of Inter-Rater Reliability*, second ed. Advanced Analytics, LLC, 2010.

[67] Gyllstrom, K., and Moens, M.-F. Clash of the typings: finding controversies and children's topics within queries. In *Proceedings of the 33rd European Conference on IR Research* (2011), ECIR '11, Springer-Verlag, pp. 80–91.

[68] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. The weka data mining software: An update. *SIGKDD Explor. Newsl. 11*, 1 (Nov. 2009), 10–18.

[69] Halvey, M. J., and Keane, M. T. Exploring social dynamics in online media sharing. In *Proceedings of the 16th International Conference on World Wide Web* (2007), WWW '07, ACM, pp. 1273–1274.

[70] Harper, F. M., Raban, D., Rafaeli, S., and Konstan, J. A. Predictors of answer quality in online q&a sites. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems* (2008), CHI '08, ACM, pp. 865–874.

[71] Hatzivassiloglou, V., and McKeown, K. A quantitative evaluation of linguistic tests for the automatic prediction of semantic markedness. In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics* (1995), ACL '95, Association for Computational Linguistics, pp. 197–204.

[72] Hatzivassiloglou, V., and McKeown, K. R. Predicting the semantic orientation of adjectives. In *Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics* (1997), EACL '97, Association for Computational Linguistics, pp. 174–181.

[73] He, B., and Ounis, I. Query performance prediction. *Inf. Syst. 31*, 7 (Nov. 2006), 585–594.

[74] He, X., Gao, M., Kan, M.-Y., Liu, Y., and Sugiyama, K. Predicting the popularity of web 2.0 items based on user comments. In *Proceedings of the 37th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2014), SIGIR '14, ACM.

[75] Hsu, C.-F., Khabiri, E., and Caverlee, J. Ranking comments on the social web. In *Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 04* (2009), CSE '09, IEEE Computer Society, pp. 90–97.

[76] Hu, M., Sun, A., and Lim, E.-P. Comments-oriented document summarization: understanding documents with readers' feedback. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2008), SIGIR '08, pp. 291–298.

[77] Hua, G., Zhang, M., Liu, Y., Ma, S., and Ru, L. Hierarchical feature selection for ranking. In *Proceedings of the 19th International Conference on World Wide Web* (2010), WWW '10, ACM, pp. 1113–1114.

[78] Jain, V., and Varma, M. Learning to re-rank: Query-dependent image reranking using click data. In *Proceedings of the 20th International Conference on World Wide Web* (2011), WWW '11, ACM, pp. 277–286.

[79] J.Kunegis, A., and C.Bauckhage. The slashdot zoo: Mining a social network with negative edges. *ACM Transactions on Intelligent Systems and Technology, 2011* (2009).

[80] Joachims, T. Text categorization with suport vector machines: Learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning* (1998), ECML '98, pp. 137–142.

[81] Joachims, T. Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2006), KDD '06, ACM, pp. 217–226.

[82] Kang, I.-H., and Kim, G. C. Query type classification for web document retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2003), SIGIR '03, ACM, pp. 64–71.

[83] Kim, S.-M., Pantel, P., Chklovski, T., and Pennacchiotti, M. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (2006), EMNLP '06, Association for Computational Linguistics, pp. 423–430.

[84] Kittur, A., Suh, B., Pendleton, B. A., and Chi, E. H. He says, she says: conflict and coordination in wikipedia. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2007), CHI '07, ACM, pp. 453–462.

[85] Kucuktunc, O., Cambazoglu, B. B., Weber, I., and Ferhatosmanoglu, H. A large-scale sentiment analysis for yahoo! answers. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining* (2012), WSDM '12, ACM, pp. 633–642.

[86] Li, Q., Wang, J., Chen, Y. P., and Lin, Z. User comments for news recommendation in forum-based social media. *Information Sciences 180*, 24 (2010), 4929–4939.

[87] Li, X., Wang, Y.-Y., and Acero, A. Learning query intent from regularized click graphs. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2008), ACM, pp. 339–346.

[88] Liu, T.-Y. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval 3*, 3 (2009), 225–331.

[89] Lu, Y., Zhai, C., and Sundaresan, N. Rated aspect summarization of short comments. In *Proceedings of the 18th international conference on World wide web* (2009), WWW '09, ACM, pp. 131–140.

[90] Macdonald, C., Santos, R. L., and Ounis, I. On the usefulness of query features for learning to rank. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (2012), CIKM '12, ACM, pp. 2559–2562.

[91] Manning, C., and Schuetze, H. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

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

[93] McCallum, A., and Nigam, K. A comparison of event models for naive bayes text classification. In *IN AAAI-98 WORKSHOP ON LEARNING FOR TEXT CATEGORIZATION* (1998), AAAI Press, pp. 41–48.

[94] Merler, M., Yan, R., and Smith, J. R. Imbalanced rankboost for efficiently ranking large-scale image/video collections. In *CVPR* (2009), pp. 2607–2614.

[95] Metzler, D., and Bruce Croft, W. Linear feature-based models for information retrieval. *Inf. Retr. 10*, 3 (2007), 257–274.

[96] Mishne, G., and Glance, N. Leave a reply: An analysis of weblog comments. In *Workshop on the Weblogging ecosystem* (2006).

[97] Mishra, A., and Rastogi, R. Semi-supervised correction of biased comment ratings. In *Proceedings of the 21st International Conference on World Wide Web* (2012), WWW '12, ACM, pp. 181–190.

[98] Mohan, A., Chen, Z., and Weinberger, K. Web-search ranking with initialized gradient boosted regression trees. *Journal of Machine Learning Research 14* (2011), 77–89.

[99] Musial, K., and Kazienko, P. Social networks on the internet. *World Wide Web Journal 16*, 1 (2013), 31–72.

[100] Orimaye, S. O., Alhashmi, S. M., and Siew, E.-G. Frequency of sentential contexts vs. frequency of query terms in opinion retrieval. In *WEBIST* (2011), J. Cordeiro and J. Filipe, Eds., SciTePress, pp. 607–610.

[101] Pak, A., and Paroubek, P. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation* (may 2010), LREC '10.

[102] Pan, S. J., Ni, X., Sun, J.-T., Yang, Q., and Chen, Z. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th International Conference on World Wide Web* (2010), WWW '10, ACM, pp. 751–760.

[103] Pang, B., and Lee, L. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr. 2*, 1-2 (Jan. 2008).

[104] Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10* (2002), EMNLP '02, Association for Computational Linguistics, pp. 79–86.

[105] Park, S., Ko, M., Kim, J., Liu, Y., and Song, J. The politics of comments: Predicting political orientation of news stories with commenters' sentiment patterns. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work* (2011), CSCW '11, ACM, pp. 113–122.

[106] Pass, G., Chowdhury, A., and Torgeson, C. A picture of search. In *Proceedings of the 1st International Conference on Scalable Information Systems* (2006), InfoScale '06, ACM.

[107] Pera, M. S., Qumsiyeh, R., and Ng, Y.-K. A query-based multi-document sentiment summarizer. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (2011), CIKM '12, ACM, pp. 1071–1076.

[108] Potthast, M., Stein, B., Loose, F., and Becker, S. Information retrieval in the commentsphere. *Transactions on Intelligent Systems and Technology (ACM TIST) 3*, 4 (Sept. 2012), 68:1–68:21.

[109] Preis, T., Moat, H. S., and Stanley, H. E. Quantifying trading behavior in financial markets using google trends. *Scientific Reports 3* (2013).

[110] Qin, T., and Liu, T.-Y. Introducing letor 4.0 datasets. *CoRR abs/1306.2597* (2013).

[111] Rokicki, M., Chelaru, S., Zerr, S., and Siersdorfer, S. Competitive game designs for improving the cost effectiveness of crowdsourcing. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, CIKM '14 (Accepted Paper).

[112] Rosenberg, A., and Binkowski, E. Augmenting the kappa statistic to determine interannotator reliability for multiply labeled data points. In *Proceedings of HLT-NAACL 2004: Short Papers* (2004), HLT-NAACL-Short '04, Association for Computational Linguistics, pp. 77–80.

[113] Rowe, M., Angeletou, S., and Alani, H. Anticipating discussion activity on community forums. In *Proceedings of the PASSAT/SocialCom* (2011), pp. 315–322.

[114] Rowe, M., Angeletou, S., and Alani, H. Predicting discussions on the social semantic web. In *Proceedings of the 8th Extended Semantic Web Conference on The Semanic Web: Research and Applications - Volume Part II* (2011), ESWC'11, Springer-Verlag, pp. 405–420.

[115] San Pedro, J., Yeh, T., and Oliver, N. Leveraging user comments for aesthetic aware image search reranking. In *Proceedings of the 21st World Wide Web Conference* (2012), WWW '12, ACM, pp. 439–448.

[116] Schapire, R., and Freund, Y. *Boosting: Foundations and Algorithms*. Adaptive computation and machine learning. MIT Press, 2012.

[117] Shen, D., Li, Y., Li, X., and Zhou, D. Product query classification. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management* (2009), CIKM '09, ACM, pp. 741–750.

[118] Shmueli, E., Kagian, A., Koren, Y., and Lempel, R. Care to comment?: Recommendations for commenting on news stories. In *Proceedings of the 21st International Conference on World Wide Web* (2012), WWW '12, ACM, pp. 429–438.

[119] Shokouhi, M., and Radinsky, K. Time-sensitive query auto-completion. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2012), SIGIR '12, ACM, pp. 601–610.

[120] Siersdorfer, S., Chelaru, S., Nejdl, W., and San Pedro, J. How useful are your comments?: analyzing and predicting youtube comments and comment ratings. In *Proceedings of the 19th International Conference on World Wide Web* (2010), WWW '10, ACM, pp. 891–900.

[121] Siersdorfer, S., Chelaru, S., San Pedro, J., Altingovde, I. S., and Nejdl, W. Analyzing and mining comments and comment ratings on the social web. *ACM Transactions on the Web 8*, 3 (July 2014), 17:1–17:39.

[122] Silvestri, F. Mining query logs: Turning search usage data into knowledge. *Found. Trends Inf. Retr. 4*, 1-2 (Jan. 2010), 1–174.

[123] Skobeltsyn, G., Junqueira, F., Plachouras, V., and Baeza-Yates, R. Resin: A combination of results caching and index pruning for high-performance web search engines. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2008), SIGIR '08, ACM, pp. 131–138.

[124] Song, Y., Zhou, D., and He, L.-w. Post-ranking query suggestion by diversifying search results. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2011), SIGIR '11, ACM, pp. 815–824.

[125] Szpektor, I., Gionis, A., and Maarek, Y. Improving recommendation for long-tail queries via templates. In *Proceedings of the 20th International Conference on World Wide Web* (2011), WWW '11, ACM, pp. 47–56.

[126] Tatar, A., Leguay, J., Antoniadis, P., Limbourg, A., de Amorim, M. D., and Fdida, S. Predicting the popularity of online articles based on user comments. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics* (2011), WIMS '12.

[127] Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., and Kappas, A. Sentiment in short strength detection informal text. *JASIST 61*, 12 (2010), 2544–2558.

[128] Thelwall, M., Sud, P., and Vis, F. Commenting on youtube videos: From guatemalan rock to el big bang. *JASIST 63*, 3 (2012), 616–629.

[129] Thomas, M., Pang, B., and Lee, L. Get out the vote: determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (2006), EMNLP '06, Association for Computational Linguistics, pp. 327–335.

[130] Tonellotto, N., Macdonald, C., and Ounis, I. Efficient and effective retrieval using selective pruning. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining* (2013), WSDM '13, ACM, pp. 63–72.

[131] Tsagkias, M., Weerkamp, W., and de Rijke, M. News comments: Exploring, modeling, and online prediction. In *Proceedings of the 32nd European Conference on IR Research* (2010), ECIR '10, Springer-Verlag, pp. 191–203.

[132] Turney, P. D. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (2002), ACL '02, pp. 417–424.

[133] Turney, P. D., and Littman, M. L. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. technical report egb-1094. Tech. rep., National Research Council Canada, 2002.

[134] Vapnik, V. N. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., 1995.

[135] Vavliakis, K. N., Gemenetzi, K., and Mitkas, P. A. A correlation analysis of web social media. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics* (2011), WIMS '11, pp. 54:1–54:5.

[136] Veloso, A., Jr., W. M., Macambira, T., Guedes, D., and Almeida, H. Automatic moderation of comments in a large on-line journalistic environment. In *Proceedings of the International Conference on Weblogs and Social Media* (2007), ICWSM '07.

[137] Vuong, B.-Q., Lim, E.-P., Sun, A., Le, M.-T., Lauw, H. W., and Chang, K. On ranking controversies in wikipedia: models and evaluation. In *Proceedings of the 2008 International Conference on Web Search and Data Mining* (2008), WSDM '08, ACM, pp. 171–182.

[138] Vural, A. G., Cambazoglu, B. B., and Senkul, P. Sentiment-focused web crawling. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (2012), CIKM '12, ACM, pp. 2020–2024.

[139] Wang, L., Lin, J., and Metzler, D. Learning to efficiently rank. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (2010), SIGIR '10, ACM, pp. 138–145.

[140] Weber, I., Garimella, V. R. K., and Borra, E. Mining web query logs to analyze political issues. In *Proceedings of the 3rd Annual ACM Web Science Conference* (2012), WebSci '12, ACM, pp. 330–334.

[141] Weimer, M., Gurevych, I., and Mühlhäuser, M. Automatically assessing the post quality in online discussions on software. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions* (2007), ACL '07, Association for Computational Linguistics, pp. 125–128.

[142] Wilkinson, E. Climate change: Environmental issues vs leadership, 2012. Available at http://www.wateo.org/2012/01/02/climate-change-environmental-issues-vs-leadership-by-elisa-wilkinson/.

[143] Wu, F., and Huberman, B. A. How public opinion forms. In *Proceedings of the 4th International Workshop on Internet and Network Economics* (2008), WINE '08, Springer-Verlag, pp. 334–341.

[144] Yang, Y., and Pedersen, J. O. A comparative study on feature selection in text categorization. Morgan Kaufmann Publishers, pp. 412–420.

[145] Yano, T., and Smith, N. A. What's worthy of comment? content and comment volume in political blogs. In *Proceedings of the Fourth International Conference on Weblogs and Social Media* (2010), ICWSM '10.

[146] Yee, W. G., Yates, A., Liu, S., and Frieder, O. Are web user comments useful for search? In *Proceedings of the SIGIR'09 Workshop on LSDS-IR* (2009).

[147] YouTube. Viewership statistics, 2014. Available at https://www.youtube.com/yt/press/statistics.html.

[148] Zaragoza, H., Cambazoglu, B. B., and Baeza-Yates, R. Web search solved?: All result rankings the same? In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (2010), CIKM '10, ACM, pp. 529–538.

Curriculum Vitae

born on 1984/02/11 in Iasi, Romania

Since Dec. 2008	**University of Hannover/L3S Research Center** PhD Student in Computer Science
Since Dec. 2008	**University of Hannover/L3S Research Center** Junior Researcher
Mar. 2008 - July. 2008	**Vienna University of Technology, Austria** **Information Systems Institute** Erasmus Student
Sep. 2003 - Sep 2008	**Technical University Gh. Asachi Iasi, Romania** **Faculty of Automatics and Computer Science** Dipl.-Ing. in Computer Science
Sep. 1995 - Jun. 2003	**Gymnasium M. Eminescu Iasi, Romania** Baccalaureat (Abitur) Mathematics/Informatics branch