Advanced Topics in Information Retrieval

9. Social Media

Vinay Setty
(vsetty@mpi-inf.mpg.de)

Jannik Strötgen
(jtroetge@mpi-inf.mpg.de)
9.1. What is Social Media?
9.2. Tracking Memes
9.3. Opinion Retrieval
9.4. Feed Distillation
9.5. Top-Story Identification
9.1. What is Social Media?

- Content creation is supported by software (no need to know HTML, CSS, JavaScript)
- Content is user-generated (as opposed to by big publishers) or collaboratively-edited (as opposed to by a single author)
- Web 2.0 (if you like –outdated– buzzwords)

Examples:
- Blogs (e.g., Wordpress, Blogger, Tumblr)
- Social Networks (e.g., facebook, Google+)
- Wikis (e.g., Wikipedia but there are many more)
- …
Weblogs, Blogs, the Blogosphere

- Journal-like website, editing supported by software, self-hosted or as a service
- Initially often run by enthusiasts, now also common in the business world, and some bloggers make their living from it
- Reverse chronological order (newest first)
- Blogroll (whose blogs does the blogger read)
- Posts of varying length and topics
- Comments
- Backed by XML feed (e.g., RSS or Atom) for content syndication
Weblogs, Blogs, the Blogosphere

- WordPress.com
  - ~ 60M blogs
  - ~ 50M posts/month
  - ~ 50M comments/month

- Tumblr.com (by Yahoo!)
  - ~ 208M blogs
  - ~ 95B posts
  - ~ 100M posts/day

http://mybiasedcoin.blogspot.de
Twitter

- Micro-blogging service created in March ‘06
- Posts (tweets) limited to 140 characters
- 271M monthly active users
- 500M tweets/day = ~6K tweets/second
- 2B queries per day
- 77% of accounts are outside of the U.S.
- Hashtags (#atir2016)
- Messages (@vinaysetty)
- Retweets
Facebook, Twitter, LinkedIn, Pinterest, …
Challenges & Opportunities

- **Content**
  - plenty of context (e.g., publication timestamp, relationships between users, user profiles, comments, external urls)
  - short posts (e.g., on Twitter), colloquial/cryptic language
  - spam (e.g., splogs, fake accounts)

- **Dynamics**
  - up-to-date content – real-world events covered as they happen
  - high update rates pose severe engineering challenges (e.g., how to maintain indexes and collection statistics)
How do People Search Blogs?

- Mishne and de Rijke [8] analyzed a month-long query log from a blog search engine (blogdigger.com) and found that
  - queries are mostly informational (vs. transactional or navigational)
    - contextual: in which context is a specific named entity (i.e., person, location, organization) mentioned, for instance, to find out opinions about it
    - conceptual: which blogs cover a specific high-level concept or topic (e.g., stock trading, gay rights, linguists, islam)
  - contextual more common than conceptual both for ad-hoc and filtering queries
  - most popular topics: technology, entertainment, and politics
  - many queries (15–20%) related to current events
How do People Search Twitter?

- Teevan et al. [10] conducted a survey (54 MS employees), compared query logs from web search and Twitter, finding that queries on Twitter
  - are often related to celebrities, memes, or other users
  - are often repeated to monitor a specific topic
  - are on average shorter than web queries (1.64 vs. 3.08 words)
  - tend to return results that are shorter (19.55 vs. 33.95 words), less diverse, and more often relate to social gossip and recent events

- People also directly express information needs using Twitter: 17% of tweets in the analyzed data correspond to questions
What Data?

- Feeds (e.g., blog, twitter user, facebook page)
- Posts (e.g., blog posts, tweets, facebook posts)
- We’ll consider
  - textual content of posts
  - publication timestamps of posts
  - hyperlinks contained in posts
- We’ll ignore
  - other links (e.g., friendship, follower/followee)
  - hashtags, images, comments
Tasks

- **Meme tracking** - grouping of memes to track them over a period of time.

- **Post retrieval** - identifies posts relevant to a specific information need (e.g., how is life in Iceland?)

- **Opinion retrieval** - finds posts relevant to a specific named entity (e.g., a company or celebrity) which express an opinion about it.

- **Feed distillation** - identifies feeds relevant to a topic, so that the user can subscribe to their posts (e.g., who tweets about C++?)

- **Top-story identification** - leverages social media to determine the most important news stories (e.g., to display on front page).
9.1. What is Social Media?

9.2. Tracking Memes

9.3. Opinion Retrieval

9.4. Feed Distillation

9.5. Top-Story Identification
9.2. Tracking Memes

- Leskovec et al. [5] track memes (e.g., “lipstick on a pig”) and visualize their volume in traditional news and blogs

**Diagram:**

- **Demo:** [http://www.memetracker.org](http://www.memetracker.org)
Problem: Memes are often modified as they spread, so that first all mentions of the same meme need to be identified.

Construction of a phrase graph $G(V, E)$:

- Vertices $V$ correspond to mentions of a meme that are reasonably long and occur often enough.
- Edge $(u, v)$ exists if meme mentions $u$ and $v$:
  - $u$ is strictly shorter than $v$
  - Either: have small directed token-level edit distance (i.e., $u$ can be transformed into $v$ by adding at most $\varepsilon$ tokens)
  - Or: have a common word sequence of length at least $k$
- Edge weights based on edit distance between $u$ and $v$ and how often $v$ occurs in the document collection.
Meme Phrase Graph

Figure 2: Phrase graph. Each phrase is a node and we want to produce a collection of phrases deemed to occur on a single domain — inspection reveals that frequent inclusion to allow for very small mismatches between phrases. To capture these kinds of inclusion relations, relaxing the notion of sequence of the words in phrase clusters.

Our goal is to produce a contiguous subsequence of the words in phrase clusters. Then we partition this graph directed edges connect related phrases. Thus, we build the phrase graph.

We now add weights to the least edges so that each resulting connected component of the graph contains at least one node. By deleting the indicated edges we obtain the optimal solution.

We use techniques and found the current approach robust and scalable. The phrases constitute the frequency of the importance of each edge. The weight is defined so that it decreases for every pair of phrases and an included edge such that

\[ \text{weight}(p, q) = \frac{1}{L(p, q) + \epsilon} \]

where \( L(p, q) \) is the length of the overlap between phrases \( p \) and \( q \). Since all edges reflect an increase in the following for small amounts of textual mutation. Figure 1 depicts a very small portion of the phrase DAG for our data, zoomed in on the word-length of phrases we consider, and a lower bound on their frequency — the number of occurrences in the full corpus. We also eliminate phrases for which at least an

\[ \text{bound} \]

The phrase graph.

To begin, we define some terminology. We will refer to each item as a news article or blog post as an

\[ \text{item} \]

that occurs in one or more items as a

\[ \text{quote} \]

Our quote data is excerpting — when phrase

\[ \text{quote} \]

is so imperfect, imperfect enough that he's palling around with terrorists who would target their own country. The arrows indicate the (approximate) inclusion of one variant in another, as part of the methodology developed in Section 2.

By deleting the indicated edges we obtain the optimal solution. Thus, we build the phrase graph. We experimented with various other natural language processing techniques, or external data to create implementation or there is at least a

\[ \text{M} \]

- word consecutive overlap between

\[ \text{p}, \text{q} \]

with this property are exclusively produced by spammers.

spammers.

Thus, we build the phrase graph.

After this pre-processing, we build a graph from all phrases in the cluster should flow into a single root node that they can all be explained as "belonging" either to a single long

\[ \text{M} \]

or to a collection of phrases. Thus, we build the phrase graph.

We now add weights to the least edges so that each resulting connected component of the graph contains at least one node. By deleting the indicated edges we obtain the optimal solution. The phrases constitute the importance of each edge. The weight is defined so that it decreases for every pair of phrases and an included edge such that

\[ \text{weight}(p, q) = \frac{1}{L(p, q) + \epsilon} \]

where \( L(p, q) \) is the length of the overlap between phrases \( p \) and \( q \). Since all edges reflect an increase in the following for small amounts of textual mutation. Figure 1 depicts a very small portion of the phrase DAG for our data, zoomed in on the word-length of phrases we consider, and a lower bound on their frequency — the number of occurrences in the full corpus. We also eliminate phrases for which at least an

\[ \text{bound} \]

The phrase graph.

To begin, we define some terminology. We will refer to each item as a news article or blog post as an

\[ \text{item} \]

that occurs in one or more items as a

\[ \text{quote} \]

Our quote data is excerpting — when phrase

\[ \text{quote} \]

is so imperfect, imperfect enough that he's palling around with terrorists who would target their own country. The arrows indicate the (approximate) inclusion of one variant in another, as part of the methodology developed in Section 2.

By deleting the indicated edges we obtain the optimal solution. Thus, we build the phrase graph. We experimented with various other natural language processing techniques, or external data to create implementation or there is at least a

\[ \text{M} \]

- word consecutive overlap between

\[ \text{p}, \text{q} \]

with this property are exclusively produced by spammers.
Phrase Graph Partitioning

- Phrase graph is an directed acyclic graph (DAG) by construction
- Partition $G(V, E)$ by deleting a set of edges having minimum total weight, so that each resulting component is single-rooted
- Phrase graph partitioning is $NP$-hard, hence addressed by greedy heuristic algorithm
Applications

- Clustering of meme mentions allows for insightful analyses, e.g.:
  - volume of meme per time interval
  - peak time of meme in traditional news and social media
  - time lag between peak times in traditional news and social media
9.1. What is Social Media?

9.2. Tracking Memes

9.3. Opinion Retrieval

9.4. Feed Distillation

9.5. Top-Story Identification
9.3. Opinion Retrieval

- Opinion retrieval finds posts relevant to a specific named entity (e.g., a company or celebrity) which express an opinion about it.

- **Examples:** (from TREC Blog track 2006)
  - macbook pro
  - jon stewart
  - whole foods
  - mardi gras
  - cheney hunting

- **Title:**
  whole foods

- **Description:**
  Find opinions on the quality, expense, and value of purchases at Whole Foods stores.

- **Narrative:**
  All opinions on the quality, expense and value of Whole Foods purchases are relevant. Comments on business and labor practices or Whole Foods as a stock investment are not relevant. Statements of produce and other merchandise carried by Whole Foods without comment are not relevant.

- **Standard retrieval models** can help with finding relevant posts; but how to determine whether a post expresses an opinion?
Opinion Retrieval Task Example

<top>
<num> Number: 863
<title> netflix
<desc> Description:
Identify documents that show customer opinions of Netflix.
<narr> Narrative:
A relevant document will indicate subscriber satisfaction with Netflix. Opinions about the Netflix DVD allocation system, promptness or delay in mailings are relevant. Indications of having been or intent to become a Netflix subscriber that do not state an opinion are not relevant.
</top>
Opinion Dictionary

‣ What if we had a **dictionary of opinion words**? (e.g., like, good, bad, awesome, terrible, disappointing)

‣ Lexical resources with word sentiment information
  ‣ SentiWordNet ([http://sentiwordnet.isti.cnr.it/](http://sentiwordnet.isti.cnr.it/))

‣ General Inquirer ([http://www.wjh.harvard.edu/~inquirer/](http://www.wjh.harvard.edu/~inquirer/))

‣ OpinionFinder ([http://mpqa.cs.pitt.edu](http://mpqa.cs.pitt.edu))
He et al. [4] construct an opinion dictionary from training data

- consider only words that are neither too frequent (e.g., and, or) nor too rare (e.g., aardvark) in the post collection $D$

- let $D_{rel}$ be a set of relevant posts (to any query in a workload) and $D_{relopt} \subset D_{rel}$ be the subset of relevant opinionated posts

- two options to measure opinionatedness of a word $v$
  - **Kullback-Leibler Divergence**
    \[
    op_{KLD}(v) = P[v \mid D_{relopt}] \log_2 \frac{P[v \mid D_{relopt}]}{P[v \mid D_{rel}]}
    \]
  - **Bose Einstein Statistics**
    \[
    op_{BO}(v) = tf(v, D_{relopt}) \log_2 \frac{1 + \lambda}{\lambda} + \log_2(1 + \lambda)
    \] with \[\lambda = \frac{tf(v, D_{rel})}{|D_{rel}|}\]
He et al. [4] measure opinionatedness of a post \( d \) as follows:

- Consider the set \( Q_{\text{opt}} \) of \( k \) most opinionated words from the dictionary.
- Issue \( Q_{\text{opt}} \) as a query (e.g., using Okapi BM25 as a retrieval model).
- The retrieval status value \( \text{score}(d, Q_{\text{opt}}) \) measures how opinionated \( d \) is.

Posts are ranked in response to query \( Q \) (e.g., whole foods) according to a (linear) combination of retrieval scores:

\[
\text{score}(d) = \alpha \cdot \text{score}(d, Q) + (1 - \alpha) \cdot \text{score}(d, Q_{\text{opt}})
\]

with \( 0 \leq \alpha \leq 1 \) as a tunable mixing parameter.
Huang and Croft [5] expand the query with query-independent ($Q_I$) and query-dependent ($Q_D$) opinion words; posts are then ranked according to

$$score(d) = \alpha \cdot score(d, Q) + \beta \cdot score(d, Q_I)$$
$$+ (1 - \alpha - \beta) \cdot score(d, Q_D)$$

with $0 \leq \alpha, \beta \leq 1$ as a tunable mixing parameters and retrieval scores based on language model divergences

- Query-independent opinion words are obtained as
  - seed words (e.g., **good**, **nice**, **excellent**, **poor**, **negative**, **unfortunate**, …)
  - most frequent words in opinionated corpora (e.g., movie reviews)
Sentiment Expansion (Query Independent)

- **Examples**: (of most frequent words in different corpora)
  - **Cornell movie reviews**: like, even, good, too, plot
  - **MPQA opinion corpus**: against, minister, terrorism, even, like
  - **Blog06(op)**: like, know, even, good, too
- **Observation**: Query-independent opinion words are very general (e.g., like, good) or specific to the corpus (e.g., minister, terrorism)
Sentiment Expansion (Query Dependent)

- Query-dependent opinion words are obtained as words that frequently co-occur with query terms in pseudo-relevant documents (following the approach by Lavrenko and Croft [6])

- Given a query $q$, identify the set of $R$ of top-$k$ pseudo-relevant documents, and top-$n$ words having highest probability

\[
P[w | R] \propto \sum_{d \in R} P[w | d] \prod_{v \in q} P[v | d, w]
\]

\[
P[v | d, w] = \begin{cases} 
\frac{tf(v, d)}{\sum_u tf(u, d)} & : w \in d \\
0 & : \text{otherwise}
\end{cases}
\]

with parameter set as $k = 5$ and $n = 20$ in practice
Sentiment Expansion

Examples: (of query-dependent opinion words)

- mozart → (like, good, too, even, death, best, great, genius)
- allianz → (best, premium, great, value, traditional, fidelity)
- wikipedia → (like, open, good, know, free, great, knowledge)
Outline

9.1. What is Social Media?
9.2. Tracking Memes
9.3. Opinion Retrieval
9.4. Feed Distillation
9.5. Top-Story Identification
9.4. Feed Distillation

- Feed distillation identifies feeds (e.g., blogs, Twitter users) that are relevant to a specific (typically rather broad) topic.

- **Examples:** (from TREC Blog track 2007)
  - movie review
  - firearm control
  - baseball
  - garden
  - mobile phone

- **Challenges:** How to capture whether a blog consistently covers the given topic? How to bridge vocabulary gap to posts?
Weerkamp et al. [11] develop two approaches to feed distillation estimating language models for entire blog(ger)s and individual posts, respectively.

Notation:

- A blog $b$ is a set of posts; $|b|$ is the number of posts by $b$
- A post $p$ is a bag of terms
- $tf(v, p)$ denotes the term frequency of term $v$ in post $p$
- $B$ denotes a virtual post concatenating all posts from all blogs
Blogger Model (BM)

- Estimates a language model for each blog(ger) b

$$P[q | \theta_b] = \prod_{v \in q} P[v | \theta_b]^{tf(v,q)}$$

- Smooths probability estimates using the collection of blogs B

$$P[v | \theta_b] = (1 - \lambda_b) \cdot P[v | b] + \lambda_b \cdot P[v | B]$$

with blog-specific smoothing parameter

$$\lambda_b = \frac{\beta}{(1/|b| \cdot \sum_{p \in b} \sum_v tf(v,p)) + \beta}$$

thus smoothing blogs with shorter posts more aggressively
Blogger Model

- **Two-step generation of term** $v$ **from blog** $b$

$$P[v | b] = \sum_{p \in b} P[v | p, b] P[p | b]$$

assuming **conditional independence** of terms given blog

$$P[v | b] = \sum_{p \in b} P[v | p] P[p | b]$$

2. Draw term from post  
1. Draw post from blog

- **Uniform probability of posts** given blog (i.e., equal importance)

$$P[p | b] = 1/|b|$$

- **Maximum-likelihood estimate**

$$P[v | p] = \frac{tf(v, p)}{\sum_w tf(w, p)}$$
Posting Model (PM)

- Estimates a language model for each individual post $p$

\[ P[v \mid \theta_p] = (1 - \lambda_p) \cdot P[v \mid p] + \lambda_p \cdot P[v \mid B] \]

with post-specific smoothing parameter

\[ \lambda_p = \frac{\beta}{(\sum_w tf(w, p)) + \beta} \]

thus smoothing short posts more aggressively

- Maximum-likelihood estimate

\[ P[v \mid p] = \frac{tf(v, p)}{\sum_w tf(w, p)} \]
Posting Model

- Likelihood of generating query $q$ from language model of post $p$

$$P[q | \theta_p] = \prod_{v \in q} P[v | \theta_p] \cdot tf(v, q)$$

- Two-step generation of query $q$ from blog $b$

$$P[q | b] = \sum_{p \in b} P[q | \theta_p] \cdot P[p | b]$$

- Uniform probability of posts given blog (i.e., equal importance)

$$P[p | b] = \frac{1}{|b|}$$

1. Draw post from blog
2. Generate query from post
Query Expansion for Vocabulary Gap

- Elsass et al. [3] proposed the highly similar Large Document Model (~BM) and Small Document Model (~PM) approaches.
- Focus on bridging the vocabulary gap between high-level topic descriptions (e.g., garden) and posts (e.g., seed, flower, crop).
- Query expansion with terms from pseudo-relevant documents retrieved from different corpora:
  - Blogs (MAP 0.266 compared to small document model 0.315)
  - Posts (MAP 0.282)
  - Wikipedia articles (MAP 0.314)
  - Wikipedia passages (MAP 0.313)

NO IMPROVEMENT!
Query Expansion for Vocabulary Gap

- Query expansion based on anchor phrases in Wikipedia
  - issue original query $q$ against Wikipedia articles as corpus
  - consider top-$k$ and top-$n$ ($k < n$) results returned by query
  - score every anchor phrase $a$ occurring in any top-$n$ result and pointing to a document $d$ from the top-$k$ result as

  $$score(a) = \sum_{(a,d)} (k - rank(d))$$

  anchor phrase $a$ from top-$n$ article pointing to top-$k$ article $d$

  favoring frequent anchor phrases pointing to highly ranked articles
  
- expand query with top-$m$ anchor phrases (MAP 0.361)
Outline

9.1. What is Social Media?
9.2. Tracking Memes
9.3. Opinion Retrieval
9.4. Feed Distillation
9.5. Top-Story Identification
Thousands of news articles generated each day!
Google News

News

Top Stories
News near you
World

World

At least 3 dead in bomb attack during Eid prayers in Bangladesh

NEW DELHI - Assassins armed with crude bombs and sharp weapons attacked a security checkpoint at a large prayer gathering of Muslims in Bangladesh on Thursday, killing at least three people, police said.

Bomb blasts, gun battle shatter peace at massive Eid prayer

Militant attack on Bangladesh Eid festival kills three, wounds 14

Local Source: Sholokia explosion eyewitness talks of two consecutive explosions

Bangladesh News 24 hours

In Depth: Terror returns to Bangladesh on Eid: Four, including 3 cops, killed; one terrorist captured alive

Firepost

Live Updating: Live: Terror Strikes Bangladesh On Eid; Attackers Hided Up In School

NDTV

The Iraq War is still being fought in London, Washington and Baghdad

The much-anticipated release of the Chilcot report - the findings of a seven-year investigation into the British involvement in the 2003 U.S.

British Politics Gives a Sense of Government by Old School Chums

David Cameron, left, and Michael Gove in 2010. After taking opposite sides on the “Brexit” issue, the two old friends are reportedly no longer speaking.

US Sanctions Target North Korean Leaders for Human Rights Abuses

July 07, 2016 5:20 AM. SEOUL — The newly announced U.S. sanctions on North Korean leader Kim Jong Un for human rights violations is not expected to have any immediate impact, but advocates say the new measures will increase pressure on the ...
News Aggregators

Portal:Current events

Yahoo! News

Reddit

Flipboard

News360

Newstify
Disasters and accidents

- **Super Typhoon Nepartak**
  - The first major typhoon of 2016 threatens Taiwan, China and northern Luzon, Philippines. Thousands of people have been evacuated in Taiwan. ([The Weather Channel](https://www.weather.com), [ABC News](https://www.abcnews.com))
  - Typhoon Nepartak is expected to make landfall on mainland China on Friday and will make flooding worse. Nearly 200 people have died in flood waters in China in the past week with 41 people missing, 1.6 million relocated and almost 50000 houses collapsed. ([The Telegraph](https://www.telegraph.co.uk))

Law and crime

- A group of suspected radical Islamists hurl homemade bombs at police officers in the Kishoreganj District in central Bangladesh killing at least one officer and injuring several others. ([AP via ABC News](https://www.ap.org))

Politics and elections

- **Australian federal election, 2016**
  - Australian Prime Minister Malcolm Turnbull's Liberal/National coalition, behind Bill Shorten's Labor Party in the first 48 hours following Saturday's election, is now ahead of Labor in the Lower House, 74-71 seats, just two seats shy of the minimum needed to form a government. Minor parties and independents have won five seats; mail-in and absentee votes are still being counted. Turnbull is on the road today seeking support from a small handful of independent and small party lawmakers. ([Reuters](https://www.reuters.com), [The Australian](https://www.theaustralian.com), [Daily Mail](https://www.dailymail.co.uk))
Top-Story Identification

- **Top-story identification** (another task within the TREC Blog track) aims to identify the *most important news stories for a specific day d* based on their coverage in the blogosphere.
  - **real-time** (online, limited statistics, time critical: small lag)
  - **retrospective**: (offline, full statistics)

- **Notation:**
  - d denotes the day of interest
  - \( B_d \) is the set of posts published at day d; \( p \) denotes a post
  - n denotes a news article (consisting of headline and content)
  - \( tf(v,p) \) is the term frequency of term v in post p
Lee and Lee [7] address retrospective top-story identification using language models estimated from news and blogs.

Intuition: “News article important if discussed by many posts”

\[
\text{Importance}(n, d) \propto KL(\theta_n || \theta_{B_d})
\]

LM representing news article \( n \) and LM representing posts published at day \( d \)

(Note: This is a simplified version of the approach described in [7])

Only articles published \(-1/+1\) around the day of interest \( d \) are considered as candidates and ranked by the approach.
Top-Story Identification Workflow

1. Search Engine (eq. 9)
   - Query: News headline
   - Search results

2. Blog filter based on publication day
   - Filtered search results (Blog posts)
     - Rank 1
     - Rank 2
     - Rank 10
     - Rank n

3. Language model estimation from blogs
   - Language model estimation from article

4. Interpolation with $\beta$
   - $\theta_{LM}^{a+b}$
Language model for blog posts published at $d$ is estimated as

$$
P[v \mid \theta_{B_d}] = \frac{tf(v, B_d) + \mu \cdot \frac{tf(v,B)}{\sum_w tf(w,B)}}{\left(\sum_w tf(w, B_d)\right) + \mu}
$$

using Dirichlet smoothing with the collection of all posts $B$.
News-Story Language Model

- **Option 1**: Estimate directly from content of news article
  \[ P[v \mid \theta_n] = \frac{tf(v, n) + \mu \cdot \frac{tf(v, N)}{\sum_w tf(w, N)}}{\sum_w tf(w, n)} + \mu \]
  using Dirichlet smoothing with the entire news collection $N$

- **Option 2**: Estimate from top-k pseudo-relevant blog posts $B_n$ retrieved using headline as query and published within -1/+1 month of the news article; again using Dirichlet smoothing with the collection of all posts $B$

- **Option 3**: Interpolate language models estimated from news article content and top-k pseudo-relevant blog posts
Summary

- Meme tracking
  grouping variants of memes to track them over time

- Opinion retrieval
  finds posts expressing an opinion about a specific named entity

- Feed distillation
  identifies feeds worth following for a given high-level topic

- Top-story identification
  spots most important news articles based on coverage in blogs

- Vocabulary gaps
  are a common obstacle in IR but can often be bridged

- Language models
  are versatile and can be used to address many (if not most) tasks
References

Time is of the Essence: Improving Recency Ranking Using Twitter Data,
WWW 2010

[2] M. Efron:
Information Search and Retrieval in Microblogs,

Retrieval and Feedback Models for Blog Feed Search,
SIGIR 2008

[4] B. He, C. Macdonald, J. He, Iadh Ounis:
An Effective Statistical Approach for Blog Post Opinion Retrieval,
CIKM 2008

[5] X. Huang and W. B. Croft:
A Unified Relevance Model for Opinion Retrieval,
CIKM 2009

[6] V. Lavrenko and W. B. Croft:
Relevance-Based Language Models,
SIGIR 2001
Identifying top news stories based on their popularity in the blogosphere,
Information Retrieval 17:326–350, 2014

[8] G. Mishne and M. de Rijke:
A Study of Blog Search,
ECIR 2006

Information Retrieval on the Blogosphere,

#TwitterSearch: A Comparison of Microblog Search and Web Search,
WSDM 2011

Blog feed search with a post index,
Information Retrieval 14:515–545, 2011