Flávio Nuno Fernandes Martins

Master of Computer Engineering

Temporal Information Models for Real-Time Microblog Search

Thesis submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy in
Computer Science

Adviser: João Magalhães, Associate Professor, NOVA University Lisbon

Co-adviser: Jamie Callan, Full Professor, Carnegie Mellon University

Examination Committee

Chair: Prof. Dr. Pedro Barahona
Rapporteurs: Prof. Dr. Krisztian Balog
Prof. Dr. Djoerd Hiemstra
Members: Prof. Dr. Mário J. Silva
Prof. Dr. João Magalhães
Prof. Dr. Nuno Preguiça

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

November, 2018

Temporal Information Models for Real-Time Microblog Search

Created using the (pdf)LaTeX processor and the "novathesis" template[1], developed at the Dep. Informática of FCT-NOVA [2].
[1] https://github.com/joaomlourenco/novathesis [2] http://www.di.fct.unl.pt

In memory of my godmother Cecília
(1924 – 2013)

Acknowledgements

First and foremost, I would like to express my sincerest gratitude to my supervisory team composed by João Magalhães (Universidade Nova de Lisboa) and Jamie Callan (Carnegie Mellon University) for their unwavering support and mentorship throughout this rewarding academic journey. Also, to Mário J. Silva and Nuno Preguiça who have been so generous to follow this thesis closely starting from its early beginnings to the finish line as part of the thesis committee. Finally, I would like to thank Nuno Correia, who has helped me through his roles of coordinator of the PhD Program in Computer Science as well as coordinator of the Multimodal Systems area of NOVA LINCS.

I also acknowledge the different funding sources that made this thesis possible, this work has been partially funded by the CMU Portugal research project GoLocal Ref. CMUP-ERI/TIC/0033/2014, by the H2020 ICT project COGNITUS with the grant agreement nº 687605 and by the project NOVA LINCS Ref. UID/CEC/04516/2013.

To everyone in lab P3-13 at the DI with whom I had the pleasure to share a working space with, both in the past and more recently, I extend to you, my friends, my sincerest gratitude. A very special mention to my lab mates Filipa Peleja, André Mourão, David Semedo, Carla Viegas, and Gustavo Gonçalves for their camaraderie. On the Pittsburgh side, to the people I crossed paths with at CMU and everyone in room GHC 6418 at LTI, my fellow ephemeral Tartans, thank you all for the good times. A special thanks to Yubin Kim for she was always generous to lend her knowledge and experience whenever I asked her questions from the other side of the Atlantic.

To my siblings, Sofia, Diogo, and Catarina and to my friends and family who have supported me along the way. Finally, I want to thank my parents, Hélder and Goreti, and especially my partner Ana who was supportive in every way.

Thank you. Obrigado. 谢谢. 감사. 感謝.

“A ship in port is safe, but that is not what ships are for.
Sail out to sea and do new things.”
– Rear Admiral Grace Hopper

Abstract

Real-time search in Twitter and other social media services is often biased towards the most recent results due to the "in the moment" nature of topic trends and their ephemeral relevance to users and media in general. However, "in the moment", it is often difficult to look at all emerging topics and single-out the important ones from the rest of the social media chatter. This thesis proposes to leverage on external sources to estimate the duration and burstiness of live Twitter topics. It extends preliminary research where it was shown that temporal re-ranking using external sources could indeed improve the accuracy of results. To further explore this topic we pursued three significant novel approaches: (1) multi-source information analysis that explores behavioral dynamics of users, such as Wikipedia live edits and page view streams, to detect topic trends and estimate the topic interest over time; (2) efficient methods for federated query expansion towards the improvement of query meaning; and (3) exploiting multiple sources towards the detection of temporal query intent. It differs from past approaches in the sense that it will work over real-time queries, leveraging on live user-generated content. This approach contrasts with previous methods that require an offline preprocessing step.

Keywords: information retrieval; time-aware ranking; relevance models

Resumo

A pesquisa em tempo real no Twitter e outros serviços de social media é muitas vezes enviesada para os resultados mais recentes porque os assuntos populares são discutidos "no momento" e em geral a sua relevância é efémera tanto para os users como para os media. No entanto "no momento" é muitas vezes difícil olhar para todos os tópicos emergentes e isolar os mais importantes das restantes discussões que ocorrem nas redes sociais. Esta tese propõe a alavancagem em fontes de informação externas para estimar a duração e a *burstiness* de tópicos no Twitter em tempo real. Dá continuidade investigação preliminar que mostra que a reordenação temporal utilizando fontes externas pode de facto melhorar a precisão dos resultados. Para aprofundar este tópico, seguimos três abordagens inovadoras significativas: (1) análise de múltiplas fontes de informação que exploram dinâmicas comportamentais dos utilizadores, como as edições e os streams de visualizações das páginas da Wikipédia ao vivo, para detetar tendências de tópicos e estimar o interesse do tópico ao longo do tempo; (2) métodos eficientes de federated query expansion para a melhorar a interpretação do significado do query; e (3) exploração de múltiplas fontes para a deteção da intenção temporal do query. Difere das abordagens anteriores, no sentido em que opera em pesquisas em tempo real, aproveitando o conteúdo gerado pelos utilizadores. Esta abordagem contrasta com anteriores que exigem uma etapa de pré-processamento off-line.

Palavras-chave: information retrieval; time-aware ranking; relevance models

Contents

List of Figures

List of Tables

Acronyms

BM25	Okapi BM25
CLRM	Condensed List Relevance Models
CSI	Centralized Sample Index
EMD	Earth Mover's Distance
IDF	Inverse Document Frequency
KDE	Kernel Density Estimation
LM.Dir	Language Modeling with Dirichlet prior smoothing
LTR	Learning to Rank
MAP	Mean Average Precision
NDCG	Normalized Discounted Cumulative Gain
NTCIR	NII Testbeds and Community for Information access Research
PRF	Pseudo-Relevance Feedback
QL	Query-Likelihood
RBR	Relevance-Based Ranking
RM	Relevance Models

RM3	Relevance Models Method 3
Rprec	Precision at rank R, where R is the number of relevant documents
SBR	Size-Based Ranking
TREC	Text REtrieval Conference

Symbols

c	collection
d	document
w_d	contents of document d
$d_1 \cdots d_k$	documents
t_d	timestamp of document d
$q_1 \cdots q_n$	query terms
q	query
$\mathscr{R}$	ranked set of documents retrieved

INTRODUCTION

1

"All we have to decide is what to do with the time that is given to us."

— J.R.R. Tolkien, *The Lord of the Rings*

A networked and connected world enabled the rapid adoption of microblogs and other similar social media services. The largest microblog platform at the time of writing is Twitter, with approximately 330 million active users. Posts on Twitter have been historically limited to a maximum of 140 characters. The limit changed to 280 characters on November 7, 2017, for all languages except Japanese, Korean, and Chinese. These limits nudge users towards a language style that is more informal and telegram-like. Microblog posts (tweets) can also contain typographic marks that have acquired a special meaning, most notably, the # (hashtag) and the @ (mention), used to group messages in a topical thread and to address a post as a reply to another user.

At a first glance, it may seem that people use microblogs to talk about their daily routine or what they are currently doing. However, research has shown that conversation topics are often related to real-world events that burst on the news and other media (Java *et al.*, 2007). People use microblogs not only to comment these (major) events, often in real-time, but also to seek information. In fact, Kwak *et al.* (2010) found that the majority of trending topics on Twitter are, in essence, headline news or persistent news. The variety of information propagated through microblogs offers rich sources of evidence to understand information needs better. Some of these topics are popular and discussed across the Internet, but there are also many topics composing the heavy long tail.

The "long tail" is a long-known statistical distribution that was found to characterize several different forms of online activity. Anderson (2006) popularized the term by using it to describe the fact that bestsellers are a relatively small fraction of total consumption and that products that have a low demand

or number of sales (in the tail) can collectively comprise a market share that rivals or exceeds the relatively few products in the head. Anderson discovered that web commerce and communication follow this model and that successful online businesses make the most profit with niche products, which are seldom sold individually but are suddenly profitable when sold together.

The described phenomenon is also relevant for information retrieval. It is good to satisfy the requests for the head of the distribution, but a significant portion of requests are in the long tail. The tendency is for head topics to be general, and tail topics to be more personal. For queries in the long tail, it is often vital to identify the key terms which help in classifying the query into a segment. For instance, at the height of the award season, there will be several requests on the head of the distribution for information on the Oscars' nominees, winners, and the awards show itself.

Nonetheless, on the tail of the distribution there can be requests for information about other topics that contain overlapping query terms with the topic movies. For example, someone submitting the query "Blade Runner Oscar Pistorius" in 2018 might be looking for new information on Oscar Pistorius jail appeals. Using standard query processing techniques for query modeling could steer the query towards the most popular topic movies. Using Wikipedia for feedback to expand this query could use the article about the movie "Blade Runner", which would steer away from the intent of the query.

1.1 Microblog Search Challenges

To take advantage of the valuable information contained in microblogs, it is essential to have efficient information retrieval mechanisms that can fulfill the users' information needs. The general formulation of the information retrieval problem is: given an information need, a user first formulates it as a *query* using a sequence of keywords that the information retrieval system can process.

Given a *query*, a retrieval algorithm uses it to match documents in the *collection* and then *scores* them according to some measure of how well each

document matches the *query*. Best matching retrieval algorithms for textual search engines use heuristics such as the term frequency in the documents and term rareness in the collection to estimate the relevance of documents and rank them in descending order of relevance to the *query*. The final result is the list of documents found in the searched collection that better match the query.

Microblog search is formulated as a similar information retrieval problem as Web search, and similarly to Web search, it is also a large-scale retrieval problem. On Twitter, there are 100+ million active users that generate circa 500+ million posts daily. About 80% of users use Twitter on mobile devices generating content coordinated with ongoing events and participating in live conversations. Often, the user is trying to complete a task using a search engine, this is formalized as the *query intent*.

In Web Search, users' queries can have different intents: get to a website that can help them execute a task (transactional), to look for specific information about a topic (informational), and to get to a site they want to visit (navigational) (Broder, 2002). In contrast, Twitter queries can be categorized almost exclusively as informational.

1.1.1 Identifying Relevant Time Periods

In microblog search, *time* plays perhaps the most prominent role. Therefore, it is imperative that the design of search engines for social media incorporate the temporal dimension. Users can formulate queries that have an explicit or implicit *temporal query intent*. For instance, a user can issue the query "oscars 2018" to signal explicitly that she intends to find information about 2018. Another user could issue just "Argo wins Oscar" as the query and be looking for data from a specific time period implicitly.

It is, however, essential to distinguish between historical temporal query intents that users often explicit specify in the query and more immediate temporal query intents. In the case of microblog search, temporal query intents are usually more immediate and fine-grained due to the dynamics of social media. Teevan *et al.* (2011) found that people search Twitter to find temporally relevant

information, monitor popular trends, discover breaking news, and follow how events are developing.

More specifically, users of microblog search may wish to find the most relevant results from important subtopics from the most relevant time points of interest (Singh *et al.*, 2016). People not only search and monitor search results during an ongoing event but also produce content and issue queries related to an event in anticipation and the aftermath of an event. For instance, journalists search for events in the last couple of weeks to write pieces about the general sentiment and opinion of the crowd.

1.1.2 Vocabulary Mismatch Problem

One of the most critical aspects of microblog search is the vocabulary mismatch problem. In microblog search, text similarity retrieval models can only go so far as to assert relevance. Retrieval schemes based on language modeling have proven to be very useful in standard text collections. However, due to the reduced expressiveness of microblog posts text-only similarity models can generate many ties in the scoring of documents.

The reduced expressiveness of microblog posts, due to the 140 / 280 character limit, accentuates the vocabulary mismatch problem, since queries match only a few keywords, presenting fewer opportunities to match the query to short text documents. Moreover, queries submitted to Twitter were found to be significantly shorter than queries submitted to Web search engines (1.64 words vs. 3.08 words) (Teevan *et al.*, 2011). Therefore, microblog search is a fertile playing ground for new techniques that leverage internal as well as external metadata to improve ranking and remove ties in the results.

Microblog search is following a similar approach to the current state-of-the-art in Web search engines, which is to rank Web results by combining hundreds of different ranking signals: several text retrieval scores matching the user's query to different parts of the web page, PageRank (Page *et al.*, 1999) and other authority and popularity signals, as well as several other ranking signals. It seems that for microblog search query expansion is essential to provide a

richer description of the information need. An expanded query with additional terms that can capture query intent better is likely to retrieve more relevant documents, and lead to better document ranking. Besides, information collected from external sources could also provide better clues for query formulation (Diaz and Metzler, 2006).

1.1.3 Response-Time and Efficiency Constraints

Starting from the premise that it may be difficult to formulate a good query, search engines can rely on relevance feedback to better understand the query and improve the results. The most straightforward relevance feedback mechanisms involve explicit feedback, where the users select which results are relevant.

Pseudo-Relevance Feedback, also known as blind relevance feedback, is often preferred to explicit relevance feedback because it automates the manual part of relevance feedback, simulating it by boosting the top-ranked results, without an extended user. When using pseudo-relevance feedback information is automatically extracted from the initial search results, and a new query is issued as an expansion of the original query, although this process is not visible to the user. In explicit relevance feedback, the result of the first retrieval step is visible to the user, which provides relevance feedback on the results.

Similarly to pseudo-relevance feedback, using temporal feedback in dynamic collections can raise efficiency concerns due to the initial retrieval, therefore finding more efficient methods to extract temporal signals can improve the query response times.

However, both methods still rely on performing two queries. Having multiple retrieval steps is unequivocally less efficient. Moreover, the expanded query is longer which can also hurt the search engine performance. Long queries have a higher computing cost and potentially longer response times for the user, which can be often not tolerable by users.

1.2 Efficient Time-Aware Search in Microblogs

Despite the computational cost, multi-stage retrieval systems are increasingly the norm. Nowadays, Web search engines use state-of-the-art learning to rank algorithms in its later stages. These are necessary to achieve state-of-the-art performance while standard retrieval models are increasingly used just as a first candidate document generation step (Asadi, 2013). Therefore, to allow for a more efficient search process, it has become important to restrict this more expensive score computation to a smaller pool of documents. Thus, initial stages use fast, usually less accurate, initial retrieval algorithms or filtering to obtain a smaller candidate set of documents for ranking. In contrast to previous work, we propose to improve microblog search by combining temporal feedback, multiple external Web sources, and federated search.

Previously, the definition of microblog search and the discussion on the real-time user information needs, have set the working domain of the present thesis. Within this scenario I have identified one main research objective, namely to

to advance microblog retrieval models and architectures with temporal evidence from multiple external resources.

In particular, we aimed to research and develop methods to improve query expansion and time-aware document ranking using the following research questions as guidelines:

1. How to robustly identify the most relevant time periods for a query?
2. How to fully exploit and combine both the lexical and temporal evidence contained in external collections to improve the ranking of time-sensitive queries?
3. How to estimate relevance models for a set of feedback documents capturing both the notions of topic relevance and temporal relevance?

1.2.1 Leveraging Web Dynamics to Mine Temporal Evidence

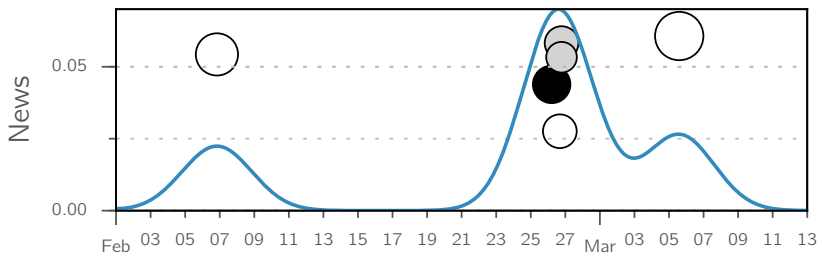


Figure 1.1: News and feedback temporal estimation.

In the past, with traditional Web search, it has been assumed that recent documents are more relevant (Li and Croft, 2003). However, this assumption has been revised for time-sensitive queries in social media search, where relevant documents tend to cluster on multiple time periods (Dakka *et al.*, 2012; Efron *et al.*, 2014). Most of the studies have typically focused on the use of a single source of information, either the corpus itself (e.g., Twitter) or an external source, such as Wikipedia.

This assumption leads us to the second research strand where novel strategies can be devised by assuming that all event trends may have an impact on the temporal signals coming from multiple Web sources. This research strand is further motivated by the increasing use of multiple sources of information for query formulation tasks (Bendersky *et al.*, 2012), without contemplating the role of temporal information.

What is occurring now is discussed in many places on the Web at the same time. The same event is retold on the Internet several times across numerous channels. This phenomenon gives us multiple sources of evidence about the same topic or event. On Twitter, conversation topics are often related to real-world events that burst on the news and other media. At the same time, related page articles on Wikipedia, the online encyclopedia, can show a higher than average number of edits and page views resulting from the interest of users towards additional information. These kinds of Web dynamics can be used to

improve ranking performance through time-aware query modeling and time-aware ranking techniques.

Figure 1.1 illustrates the power of temporal evidence arising from multiple sources for estimating information relevance. Moreover, they also pay little attention to the efficiency of the query formulation process. Sometimes relevant time periods cannot be predicted from temporal feedback alone. That is why we look for other sources that might contain cleaner temporal information.

1.2.2 Multiple Sources About Temporal Relevance

The page views statistics (Ciglan and Nørvåg, 2010) and edit history (Georgescu *et al.*, 2013; Steiner *et al.*, 2013) of Wikipedia articles have been used to detect emerging events and entities. However, to the best of our knowledge, this is the first work that explores the use of activity on Wikipedia for time-aware ranking.

Wikipedia, the most popular online encyclopedia, is continuously updated in real-time with new revisions and new articles edited by online users. Similarly to what happens in social media services, major events inspire the interest of the users towards page articles related to the subject, which can show a spike of edits and page views near the dates of an event. When changes are made to Wikipedia articles we can find what content has changed.

The news is also a good source of clean journalistic language and reliable timestamps. We found that an expansion corpus can be built in parallel to the target corpus, by crawling and indexing a selected number of accounts from Twitter itself. For instance, news outlets accounts publish posts on Twitter about the news as they happen. Their posts can be automatically assigned, using a clustering algorithm, to topic-based shards (verticals), which will then contain a topical partition (cluster) of the expansion corpus. Since only the most relevant verticals are searched this methodology can help achieve a better estimation of temporal relevance using feedback documents.

1.2.3 Multiple Sources About Query Meaning

Traditionally, the estimation of relevance models disregards the temporal dimension of relevance to compute expansion terms. Because the queries often have a *temporal query intent* to improve the estimation of relevance models it is important to estimate *temporal relevance models*. In temporal relevance models, term-selection and weighting of expansion terms incorporates the notion of temporal relevance into the estimation of the relevance model. Therefore, expansion terms from relevant time periods will be better represented in the final relevance model used for retrieval and the documents that were published in relevant time periods have a better chance to rank at the top of the final ranking, which can lead to better retrieval effectiveness.

First, it has the potential to lead to fewer instances of query-drift than standard pseudo-relevance feedback since the appropriate verticals are selected and contribute expansion terms. The use of an external expansion corpus with cleaner vocabulary can improve the quality of the query expansions. Second, only the most relevant vertical are searched and the response times for query expansion are decreased since the workload is distributed in parallel to vertical search engines.

1.2.4 Efficient Federated Search Architecture

Federated search (or distributed information retrieval) is an information retrieval architecture that employs a federation of search engines often with the objective of providing better efficiency, or as in the case of Web search, to provide a more specialized search for certain verticals. In a federated search scenario, a resource selection algorithm selects just a few of the verticals for searching. There are three main types of resource selection algorithms: sample-document, vocabulary-based, or classification based.

We propose the adaptation of the federated search architecture to provide efficient and effective pseudo-relevance feedback. Using this architecture to search the corpus used for query expansion, be it the target corpus or another external corpus, can provide additional advantages. We depart from the previous

work in federated information retrieval by also focusing on the efficiency of the query expansion process in this environment. Thus, we propose solutions that provide a balance between effectiveness and efficiency arising from

- the organization of the query expansion corpus into verticals;
- the best performing resource selection algorithms.

1.2.5 Summary of Contributions

To make progress on the previously identified microblog search challenges, we bring together federated search and temporal information retrieval techniques. First, we pursued a novel efficient query expansion architecture that can leverage external large-scale information streams about multiple topics (verticals).

In Web search, verticals are often specialized search engines to search different types of media or collections, such as Web, news, or images. In federated search, verticals are often synonymous to topical index shards. These could be created using manual curation to create topical verticals for music, movies, sports, or automatically created using a clustering algorithm.

In contrast, with the verticals from Web search, it is often assumed in federated search that the same search engine algorithm is used to search all verticals. The large volume of information on microblog streams can be a valuable resource as close-in-time documents are potentially related to each other and can share common topics. This aspect gives us multiple sources of evidence that can be aggregated in verticals to improve the retrieval efficiency.

Second, we pursued the integration of multiple sources of temporal evidence to improve the estimation of relevant time periods for a query and ultimately better time-aware ranking. We depart from previous approaches, by observing that events have an impact not only on Twitter but also on other Web sources, such as the news and Wikipedia.

The intuition is that discussions about an event (and therefore relevant documents) are more likely to occur around the same time periods across multiple Web sources. For instance, (major) events inspire the interest of users

towards related articles on Wikipedia that can observe a spike of edits and page views. Under this objective, we investigate how this kind of Web dynamics can be used to provide several independent temporal signals and broader coverage. In this context, we aim at refining the estimation of the relevant time periods for a time-sensitive query for time-aware ranking.

Finally, we consolidate the research into a novel approach with more general applicability. Leveraging both the lexical and temporal evidences from multiple external collections efficiently via the use of a federated search architecture we tackle two of the major challenges: query expansion and time-aware ranking.

1.3 Publications

The research carried out in this thesis resulted in accumulated expertise and materializes in these following main contributions to the scientific community:

1. **ACM WSDM '19 Full Paper**

   A full paper concerning the use of lexical and temporal information from verticals for query modeling and ranking in microblog search was accepted at ACM WSDM '19:

   - F. Martins *et al.* 2019. "Modeling Temporal Evidence from External Collections." In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. *WSDM '19*. Melbourne, Australia: ACM

2. **ACM ICTIR '18 Full Paper**

   A full paper concerning the use of news verticals for efficient query modeling was accepted for oral publication at ACM ICTIR '18:

   - F. Martins *et al.* 2018. "A Vertical PRF Architecture for Microblog Search." In: *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*. *ICTIR '18*. New York, NY, USA: ACM

3. **ACM WSDM '16 Full Paper**

   Our first conference paper that concerns the subjects proposed in this thesis was accepted for oral presentation at ACM WSDM '16:

   - F. Martins *et al.* 2016a. "Barbara Made the News: Mining the Behavior of Crowds for Time-Aware Learning to Rank." In: *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. *WSDM '16*. San Francisco, CA, USA: ACM

4. **ECIR '16 Demo Paper**

   An online Twitter real-time search engine was built and demoed to demonstrate the techniques developed throughout the time span of the thesis and was presented at ECIR '16:

   - F. Martins *et al.* 2016b. "Jitter Search: A News-Based Real-Time Twitter Search Interface." en. In: *Advances in Information Retrieval*. *ECIR '16*. Springer, Cham. 841–844. ISBN: 978-3-319-30670-4 978-3-319-30671-1. DOI: 10.1007/978-3-319-30671-1_77

5. **WWW '18 Companion Poster**

   A poster comparing the effectiveness of different techniques to ephemeral summarization in microblog search was presented at WWW '18:

   - G. Gonçalves *et al.* 2018. "Analysis of Subtopic Discovery Algorithms for Real-Time Information Summarization." In: *Companion Proceedings of the The Web Conference 2018*. *WWW '18*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 1855–1856. ISBN: 978-1-4503-5640-4. DOI: 10.1145/3184558.3191651

6. **Participation in the TREC Real-Time Summarization Track**

   The TREC Real-Time Summarization *track* promoted a task where a user receives a daily *email digest* that summarizes "what happened" that day with respect to the interest profiles. This task can be seen as an *ad hoc* retrieval task over a large dynamic *tweet* collection. NIST provided the test topics to participating groups. The research results obtained with our participation in this evaluation campaign were published in the following technical reports:

   - G. Gonçalves *et al.* 2017. "NOVASearch at TREC 2017 Real-Time Summarization Track." In: *Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017, Gaithersburg, Maryland, USA, November 15-17, 2017*. Ed. by E. M. Voorhees and A. Ellis. Vol. Special Publication 500-324. National Institute of Standards and Technology (NIST)

7. **Participation in the TREC Microblog Track**

   The TREC Microblog *track* promoted an *ad hoc* retrieval task over a large *tweet* collection. NIST provided access to an extensive Twitter data set and test topics to participating groups. The research results obtained with our participation in this evaluation campaign were published in the following technical reports:

   - F. Martins and J. Magalhães. 2014. "NovaSearch at TREC 2014 Microblog Track." In: *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*
   - F. Martins *et al.* 2013. "NovaSearch at TREC 2013 Microblog Track: Experiments with Reranking Using Wikipedia." In: *Proceedings of The Twenty-Second Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*

1.4 Overview of Thesis Organization

This thesis is divided into six distinct sections. The first chapter summarizes the research and introduces the basic concepts and problems approached. Chapter 2 introduces fundamental concepts of information retrieval and the related work. It begins with the background and some fundamental concepts of information retrieval, such as probabilistic retrieval, language, and relevance models. It then goes on to describe prior work in federated search, including resource selection algorithms. The related work then ends with temporal information retrieval and time-aware query reformulation. In Chapter 3 we present a real-time distributed query expansion system based on news verticals. We describe how selecting only the most relevant verticals for each query led to a more efficient query expansion process and improved retrieval effectiveness. A learning to rank approach to information retrieval in microblogs that uses temporal features extracted from multiple Web sources is introduced in Chapter 3. Chapter 5 presents techniques to detect new events in real-time on Twitter and to track these to build topic-focused streams of information. Chapter 6 summarizes the research contributions and publications, and presents directions for future work.

BACKGROUND AND RELATED WORK

2

"Identity requires
the participation of everyone else."
— Richard Ayoade

In this chapter we present the related work to contextualize the research direction. We start by describing the background and some fundamental concepts of information retrieval. Furthermore, we describe the state-of-the-art and relevant supporting work. Information Retrieval (IR) is a branch of computer science concerned with satisfying user information needs. Given an information need, the information retrieval system retrieves the information that better satisfies it in descending order of relevance.

The general formulation of the information retrieval problem is the following: given an information need, a user first formulates it as a *query* that can be understood by the information retrieval system. Given a *query*, a retrieval algorithm uses it to match documents in the *collection*, c, and then *scores* them according to some measure of how well each document matches the *query*. The final result is the list of documents, $\mathcal{R} = d_1 \cdots d_k$, found in the searched collection that better match the query.

An information need is usually formulated by issuing a query q to an information retrieval system (i.e., a sequence of keywords $q_1 \cdots q_n$). Given this formulation, retrieval algorithms in textual search engines use heuristics such as the term frequency in the content of the document w_d and term rareness in the collection to estimate the relevance of documents and rank them in descending order of relevance to the *query*.

In the query-likelihood (QL) approach to information retrieval, the probability of relevance given a document d and a query q, $P(R \mid d, q)$, is estimated using the likelihood of generating the query q given the document d, $P(q \mid d)$. This dissertation discusses novel methods that integrate the time dimension, towards

improving the estimation of the relevance of documents for time-sensitive collections. The thesis is particularly concerned with how to use temporal evidence, possibly from external collections, to provide a better estimate of the probability of relevance for a document to a query. That is, we are interested in ranking documents by their probability of relevance to a query given two components of the document: lexical, and temporal evidences, w_d, the words in the document, and t_d, the document's timestamp.

2.1 Information Retrieval in Microblogs

Microblog search is defined as an *ad hoc* information retrieval task where the retrieval unit is the microblog post. Teevan *et al.* (2011) analyzed large-scale query logs from Twitter and found that people use search primarily to find and monitor results for:

1. temporally relevant information (e.g., breaking news, real-time content, and popular trends);

2. information related to people (e.g., information about people of interest, and general sentiment and opinion).

Efron (2011) argued that microblog search shares similar challenges to blog search and that there are many parallels. He listed influence, authority and timeliness as the most important factors for ranking results in microblog search. He argued that a lot of methods previously proposed for blog search, such as the work on expert finding, incorporation of quality indicators and credibility indicators (Massoudi *et al.*, 2011), and "peopleRank" metrics can be adapted to microblog search. Efron (2011) measured the influence of the authors themselves, using features such as the number of *followers* and *retweets* as surrogates for the author's reputation, credibility, and authority.

Other approaches by Tunkeland (2009) and Weng *et al.* (2010) proposed using graph-based network authority algorithms, similarly to PageRank (Page *et al.*, 1999), to calculate credibility scores for microblog post authors. On the

other hand, Efron (2011) lists important new factors that can be used for ranking in microblogs search, such as hashtag popularity, and the @mention notation in replies. Efron (2011) proposed using the language modeling framework for information retrieval in microblogs due to their simplicity and effectiveness.

Query expansion assumed a big importance in microblog search due to the sparsity of the vocabulary in the short microblog posts. Many proposed query expansion methods take advantage of specific characteristics of microblog search. For instance, inspired by the popular use of *hashtags* to tag events or trends in microblog posts, Efron (2010) proposed learning a relevance model for each hashtag to build a hashtag feedback model for microblog search. Fan *et al.* (2015) proposed an entity feedback model that extracts keywords related to named entities contained in the query to improve the retrieval for the many queries submitted to Twitter that contain them.

One important issue in microblog search is assessing the quality and credibility of the microblog posts. Kim *et al.* (2012) argued that considerable retrieval effectiveness improvements can be obtained by simply removing spam and duplicates (i.e., *retweets*). Massoudi *et al.* (2011) argued that query expansion on microblogs should be dynamic, and include usernames, hashtags, and links. They proposed incorporating a set of *quality indicators* into the ranking, by estimating the quality of the *tweets* using mostly textual features. The features proposed include the length of the post, the presence of shouting, capitalization, emoticons, slang, unknown vocabulary, and hyperlinks. Other features include indicators tailored for microblog search such as number of *retweets*, number of *followers*, and recency. This information is incorporated by modeling the prior probability $P(d)$ for a microblog post d using a linear combination of both sets of features as a global estimate of credibility following a similar blog search approach by Weerkamp and de Rijke (2012). Naveed *et al.* (2011) proposed a similar approach where the credibility prior is modeled as the *retweet* probability. They used a logistic regression model trained with a number of quality features however, their approach is more biased towards the notion of *interestingness*.

The existence of hyperlinks in microblog posts can also be used as quality indicators. McCreadie and Macdonald (2013) proposed three different methods

to incorporate the content of hyperlinked content: virtual document integration, field-based weighting, and learning to rank. The virtual document approach favored microblog posts that contain hyperlinks, creating an imbalance and leading to degraded performance. They found that performance improved using the field-based weighting to incorporate the content of hyperlinked pages. The learning to rank model used 4 text retrieval functions to compute similarity scores between the query, the tweet-text, and the content of the hyperlinked page. In addition, *quality indicators* such as *% is stop words* and *stop word coverage*, were also added as features. The complete learning to rank model provided the best retrieval effectiveness.

Liang *et al.* (2014) frame the problem of microblog search as a rank aggregation problem. This view allows using ranked lists produced by multiple independent rankers, which boost different aspects of relevance, and aggregate them to produce the final ranking. They identified that some documents appear on just a few lists and that they could not rely on the usual heuristics. Therefore, they proposed a method to infer the scores for documents. Furthermore, they boost posts that are published around the relevant time periods based on the publishing times of documents ranked at the top of many lists.

Lin and Efron (2013a) proposed an Evaluation-as-a-Service model for assessing the effectiveness of information retrieval systems in large collections, specially social media. TREC Microblog organizers built infrastructure (Lin and Efron, 2014), via a search API for participants to submit runs without having raw access to the whole dataset, which can include sensitive data. The Evaluation-as-a-Service has been implemented in numerous evaluation campaigns (Hanbury *et al.*, 2015; Hopfgartner *et al.*, 2015; Lin and Efron, 2013b; Lin *et al.*, 2014). Voorhees *et al.* (2014) found that the diversity of Evaluation-as-a-Service runs was similar to the high-quality TREC-8. However, the evaluation-as-a-Service model is not ideal for assessing the efficiency of information retrieval systems. Methods that improve efficiency require access to low-level structures of the indexes or index the raw collection in non-anticipated ways.

The Tweets2013 collection is the most comprehensive evaluation resource for *ad hoc* retrieval on social media to date. Sequiera and Lin (2017) assessed

the suitability of evaluating the runs submitted to TREC Microblog 2013 and 2014 using an independent copy of the test dataset that is available and can be downloaded from the Internet Archive.[1] They found that 99% of the collection overlaps with the original test collection, which is accessed via the official search API and that the results are statistically indistinguishable from those evaluated with the original test collection.

2.2 Temporal Information Retrieval

Li and Croft (2003) were pioneers in the exploration of the relationship between time and relevance in information retrieval. They identified that a great portion of search queries favor more recent documents. Therefore, they tackled this problem by incorporating time into the standard query-likelihood model to balance recency with relevance. Li and Croft (2003) considered the time of the query to be the date of the most recent document and that the publishing date is available in the metadata of the documents. In recent years, the temporal aspects of information retrieval have been receiving an increased interest.

2.2.1 Time-Aware Ranking

According to Kanhabua *et al.* (2015a), existing works on time-aware ranking can be classified according to two main notions of relevance with respect to time: recency-based ranking, and time-dependent ranking. Many time-aware ranking methods leverage the language modeling framework for information retrieval.

The general language modeling approach to information retrieval (Ponte and Croft, 1998) consists in building for each document in the collection a language model M_d. If term independence is assumed, the retrieval problem is reduced to a unigram language model estimation problem. To rank documents the Bayes' rule is applied to $P(d \mid q)$ the quotient $P(q)$ is eliminated based on the

[1] https://archive.org/details/twitterstream

rank equivalence to obtain the well-known query-likelihood retrieval model:

$$P(d \mid q) = \frac{P(q \mid d) \cdot P(d)}{P(q)} \tag{2.1}$$

$$\propto P(q \mid d) \cdot P(d) \tag{2.2}$$

where $P(q \mid d)$ is the query-likelihood given a document d and $P(d)$ is a prior distribution that can be used to encode a query-independent importance of the document or uniform over all documents.

Li and Croft (2003) proposed a time-based language model to incorporate time into the standard query-likelihood model by modeling it as the prior distribution $P(d)$. To explore the assumption that in production systems, recent documents are more relevant, they replace the uniform prior probability in the original model by exponential distributions that promote documents published recently. Given document d and its timestamp t_d, they propose to model $P(d)$ in the standard query-likelihood model via the exponential distribution,

$$P(d) = \lambda e^{-\lambda(t_C - t_d)}, \tag{2.3}$$

where $\lambda \geq 0$ is the decay rate parameter of the exponential distribution and t_C is the date of the most recent document in the collection. Since the λ parameter is query-independent, globally tuned on all queries, retrieval effectiveness improves for some topics but can deteriorate for a few others since recent results are boosted equally for all queries. Nevertheless, this approach outperforms the standard query-likelihood model for retrieval in recency-biased collections.

Efron and Golovchinsky (2011) improved upon time-based language models by proposing to estimate a query-specific exponential parameter λ. Thus, they propose reranking results by calculating a maximum likelihood estimator of λ for each query from the temporal distribution of the top results. Peetz and Rijke (2013) proposed using a retention function based on a Weibull distribution for temporal document priors in microblog and news collections as an alternative to exponential distributions used in (Efron and Golovchinsky, 2011).

Jones and Diaz (2007) observed that queries have different temporal profiles. They show how to exploit the temporal profiles of queries to detect whether to ask users for feedback on the relevant time periods for queries. Jones and Diaz (2007) noted that queries that favor recency are just a subset of a broader class of time-sensitive queries. They observed that when searching any kind of timestamped documents the set of results retrieved is a timeline. The temporal profile of the query was found to be a good predictor for the mean average precision of a query.

Later models revised this assumption to handle a broader class of time-sensitive queries. Dakka *et al.* (2012) emphasized the importance of finding query-specific relevant time periods for time-sensitive queries and to integrate this information as temporal relevance in the ranking model. Therefore, temporal relevance is no longer modeled as $P(d)$ as it is now assumed to be query-specific. They devised a ranking model that explicitly divides documents into two parts: lexical, and temporal evidences, w_d, the words in the document, and t_d, the document's timestamp,

$$P(d \mid q) = P(w_d, t_d \mid q) \tag{2.4}$$

$$= P(w_d \mid q) \cdot P(t_d \mid w_d, q) \tag{2.5}$$

$$\propto P(w_d \mid q) \cdot P(t_d \mid q) \tag{2.6}$$

To estimate $P(t_d \mid q)$ Dakka *et al.* (2012) proposed using counts of documents on different time periods (bins) on a timeline. The use of histograms has several drawbacks. For instance, the selection of different time units for the width of the histogram bins can lead to dramatically different estimations, since there is no smoothing at the temporal bin boundaries. Later, Efron *et al.* (2014) overcome some of these issues by using a non-parametric method, kernel density estimation, which estimates the probability density function of a non-parametric distribution from a sample of data points. Replacing the histogram bins, by a probability density function estimated using the kernel density estimation method is linked to fewer parameters, since bandwidth selection is data-driven, (i.e., a function of the initial rank). In contrast to histograms the estimated

temporal density is smooth. Since the only hard temporal boundaries are at the edges of the timeline the estimate is continuous and smooth.

Rao *et al.* (2015) was able to reproduce the experiments by Efron *et al.* (2014) and confirmed, using a more robust experimental methodology, that this approach does indeed improve microblog retrieval effectiveness. Rao *et al.* (2017b) proposed an alternative method that uses only temporal statistics of query terms in the collection for estimating temporal relevance, which eliminates the need for an initial retrieval for feedback. Looking up temporal term statistics can be much faster since these can be stored and compressed during indexing (Rao *et al.*, 2016). Although, this approach yielded similar effectiveness to temporal feedback methods, when combined, their improvements seems to be additive and yielded the best results. Therefore, they conclude that temporal term statistics cannot yet capture the temporal signal obtained via temporal feedback from an initial retrieval.

Chen *et al.* (2018) proposed to extend document-level temporal feedback using a word-level temporal predictor in order to capture more fine-grained temporal information. As they need the initial retrieval for temporal feedback they estimated the temporal relevance of words that using the temporal distribution of words in feedback documents. They incorporate this information for time-aware ranking in microblogs by adding the word-level temporal relevance information on top of temporal feedback from an initial retrieval. The word temporal relevance was also beneficial for pseudo-relevance feedback. Incorporating this method by re-ranking the initial retrieval used for estimating the relevance model and then re-rank after the final retrieval outperformed using only document-level temporal feedback.

Rao *et al.* (2017a) examined the effectiveness of neural ranking models for ranking microblogs integrating lexical and temporal signals. The neural ranking approaches evaluated did not improve retrieval effectiveness significantly compared to the query-likelihood baseline. However, the neural ranking model is better, at the top ranks when combined with temporal signals, than the strong KDE-based baselines Efron *et al.*, 2014.

2.2.2 Learning to Rank for Time-Aware Ranking

According to Kanhabua *et al.* (2015a), previous work on time-aware ranking has followed one of two main approaches:

- a mixture model combining textual similarity and temporal similarity
- a probabilistic model generating a query from the textual and temporal part of a document independently

Mixture model methods can be seen as linear models combining feature scores with learned weights using a learning to rank machine learning algorithm such as Coordinate ascent. There are several previous works employing feature-based methods and machine learning for ranking with temporal features.

Recently, Kanhabua *et al.* (2015b) studied the problem of detecting event-related queries in Web search streams (query logs). Dai *et al.* (2011) proposed to run each query against a set of rankers to try to minimize the risk of degraded performance due to misclassifying the query in terms of recency intent. Elsas and Dumais (2010) studied how the temporal dynamics of web content, the frequency of changes, relates to relevance and how it can be leveraged in ranking algorithms. To explore this correlation they designed a probabilistic ranking model that attributes different weights to terms based on their temporal characteristics. Furthermore, they incorporate the rate of change in documents in a query-independent document prior to favor dynamic documents. Since documents change, having a static view of the content at one point is not ideal.

More recent time-dependent ranking approaches have resorted to learning to rank techniques that exploit non-temporal and temporal features (Kanhabua and Nørvåg, 2012). For instance, Costa *et al.* (2014) studied how long-term web document persistence relates to relevance. Relevant Web documents were found to persist over longer periods of time, over multiple web snapshots, and to accumulate more revisions. To explore this correlation they designed novel temporal features and integrated them in ranking using learning to rank algorithms. Furthermore, since Web archives can span over decades of web pages they proposed a temporal-dependent learning to rank framework to account for

different practices and characteristics in the web over time. Their approach is to learn a model for each time period of the web archive, which is more effective for retrieval on this specific period than a single model.

2.2.3 Temporal Expressions

Most of the work on time-aware ranking focuses on the publication times of documents. However, in some collections it is not always possible to determine the publishing time of documents. For example, when crawling Web pages you may not be able to extract the publication date of the document and may only be able to get an approximated date by looking up the first time this page was crawled. In order to determine the time of non-timestamped documents, temporal expressions can be extracted from the content of the document.

Alonso *et al.* (2007) highlighted the importance of leveraging the temporal information embedded in documents in the form of temporal expressions to enhance the functionality of current information retrieval applications. Schilder and Habel (2001) identified three temporal expression categories: *explicit*, those that can be mapped directly to a date, *implicit*, those that need to be further resolved using an external engine, and *relative*, expressions that can be mapped given a known reference time point. They propose recognizing such temporal expressions and other types of temporal information embedded in documents to improve document ranking and develop other time-related functionality. Jatowt *et al.* (2013) notes however that temporal expressions in the document can refer to two main types of temporal information: *document content time*, the publishing time of the document, e.g., in the byline of a news article, or *document focus time*, the temporal focus of the document, e.g., the dates of the event described in a news story.

Berberich *et al.* (2010) proposed the use of a language model retrieval framework (i.e., query-likelihood) to match explicit temporal expressions in queries to temporal expressions contained in the documents. Kanhabua and Nørvåg (2010) extend this work for implicit queries by estimating the most relevant time-interval for the query using the top-ranked documents retrieved

in a feedback fashion. In their work, they considered using either temporal expressions contained in the documents as well as document publishing times.

2.2.4 Temporal Web Dynamics

Many previous studies have explored the dynamics of content changes on the Web. Web pages are dynamic in the sense that they can change over time. Traditionally, pages can change due to edits done by page authors. However, with the advent of collaborative Web platforms and Web 2.0 more websites allow changes by users and the contribution of user-generated content. One categorization of content changes considers two classes:

- Dynamic and non-versioned – (e.g., news and social media)
- Dynamic and versioned – (e.g., collaborative platforms with revisions)

Intuitively, pages with recent editing activity and that have been significantly changed are more relevant. Jatowt *et al.* (2005) proposed exploiting this intuition by analyzing the changed contents in documents. The relevance of the changes with respect to a query is estimated using a text similarity score and the size of each change is recorded. This is combined with information about the frequency of the changes to rank document that have been significantly modified recently.

Link-based web ranking algorithms such as PageRank (Page *et al.*, 1999) are often calculated using a single web crawl snapshot. Dai and Davison (2010) proposed a time-sensitive version of PageRank based on multiple web crawls at different points in time to incorporate page freshness, a measure of how fresh the page content is from content changes, and in-link freshness, a measure of how much other pages linked to the page recently. They design a temporal random surfer model and incorporate this information into authority propagation to favor fresh pages. Thus, pages that were popular a long time ago and that have since become stale are no longer ranked higher. Experiments on an archival web corpus from the Internet Archive[2] have shown that incorporating time this

[2] http://www.archive.org

approach outperforms PageRank in NDCG@5 on both relevance and freshness by 17.8% and 13.5%, respectively.

Text streams are increasingly ubiquitous and are the source of a large volume of messages that have semantic as well as temporal information. Text mining over multiple text streams indexed by the same set of time points (coordinated text streams) can uncover interesting latent associations and (major) events behind topic patterns that burst simultaneously in multiple streams, a correlated bursty topic pattern. Wang *et al.* (2007) propose the use of a probabilistic mixture model with PLSA (Hofmann, 1999) to identify bursty patterns and their bursty time periods from coordinated streams simultaneously using temporal information. It aligns topics from multiple streams correlating the time distribution of the detected topics, even if the streams don't share the same vocabulary or language.

2.2.5 Temporal Queries

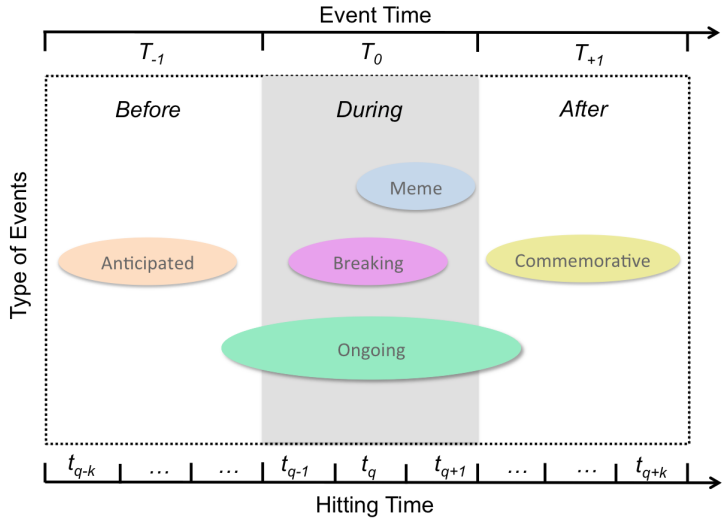


Figure 2.1: Latest taxonomy of temporal queries. (Kanhabua *et al.*, 2016)

Several works have studied the temporal intent of queries and have developed different categorizations for queries with respect to time. In Figure 2.1 we can visualize one of the latest taxonomies of temporal queries, proposed by

Kanhabua *et al.* (2016), it categorizes queries into six classes: *anticipated*, *breaking*, *commemorative*, *meme*, *ongoing*, and *atemporal*. Temporal expressions can appear in documents as well as in the queries.

Nunes *et al.* (2008) found that only approximately 1.5% of general web queries contain an *explicit* temporal expression by using a temporal expression tagger on query logs. Notice that there are specific domains where temporal expressions are used more frequently: news, sports, politics, etc. In addition, queries from expert users such as historians, journalists, or social scientists are expected to have temporal information needs more frequently.

Metzler *et al.* (2009) analyzed query logs to estimate that 7% of general web queries are implicitly year qualified queries and thus have an implicit temporal intent. Implicitly year qualified queries are not explicitly qualified with a year but the user seems to have formulated the query with a specific year in mind. They proposed mining these queries from query logs and developed a method to bias ranking functions to favor documents matching the user's *implicit* temporal query intent. Some queries might appear to be temporal since they are often qualified with a year. However, there might be other formulations that are not temporal. They define this phenomenon using the term *temporal ambiguity*.

Zhang *et al.* (2010) performed query log analysis and classified the queries into three categories: *explicit*, *implicit*, and *no timestamp*. They found that 13.8% of web queries contain explicit timestamps. To find implicit temporal queries they looked for queries that do not contain explicit timestamps but have explicit counterparts. They estimate that there are about 17.1% of temporally implicit queries. The remaining 69.1% queries are classified as *no timestamp*.

Web search engines started integrating recency into its search results to rank recency queries. Sometimes time-aware ranking methods can degrade the performance of non-temporal queries, thus recency query classification became important. This poses an additional challenge (namely *temporal query performance prediction*), which is yet to be resolved in time-aware retrieval, since deciding when not to use temporal signals is a hard task.

Previous works have tried to mitigate this problem in various ways. Diaz (2009) determined the newsworthiness of a query by predicting the probability

of a user clicking-through a news result given a query. König *et al.* (2009) estimated the click-through rates on news search results to select when whether to show news results interleaved with web search results for a query. Similar query recency classifiers were later used to select whether to apply temporal ranking for a given query. Dong *et al.* (2010) proposed a ranking model based on learning to rank to incorporate multiple recency features into web search results ranking. They showed that in some cases applying this ranking model to non-time-sensitive queries can actually degrade retrieval effectiveness. Therefore, they built a query classifier focusing on the detection of breaking-news queries, claimed to be about 1-2% of the web search engine's queries.

Chang *et al.* (2013) proposed to use Twitter data in the ranking of results for breaking-news queries. They propose using posts crawled from microblogs to discover new fresh URLs at a faster rate than re-crawling the Web. Only about 10% of the fresh URLs on the Twitter stream had been already crawled once by the web crawler. Moreover, for very fresh documents ranking models cannot rely on link-based or click-based authority metrics since these have just a few, if any. Using the automatic classifier proposed in Dong *et al.* (2010) they use multiple features to rank very recent Twitter URLs and improve recency ranking for breaking-news queries.

Gupta and Berberich (2014) investigated how to identify time intervals of interest to a query using both the publication dates and the temporal expressions contained in pseudo-relevant documents. They showed that instantiations of a generative model that considers the uncertainty inherent to temporal expressions was more effective. Campos *et al.* (2017) proposed considering the different time periods that are relevant to the query to reach a better temporal similarity score between the query and the set of extracted temporal expressions. They use a classification approach that leverages the co-occurrence of temporal expressions and words in the collection of documents to build a classifier to classify the relevance of temporal expressions towards a query.

The methods discussed earlier consider an information need that is expressed using a query. When a query does not exist and the user simply wishes to find what is happening now, Buntain *et al.* (2016) proposed leveraging a time-series

algorithm and several temporal features to find unexpected key moments from Twitter's public stream. User search behavior can be influenced by external factors such as trending news. To study the influence of trending news on user search behavior Karmaker Santu *et al.* (2017) proposed a method that identifies queries triggered by events using information mined from trending event news and query logs.

Vlachos *et al.* (2004) classified queries based on their burst shapes and burst duration on query logs. They also identify queries using similarity by matching bursts: query-by-bursts. Some works have used time-series analysis to capture temporal dynamics. The features extracted from time-series data are seasonality, trend, and surprise (Jones and Diaz, 2007; Radinsky *et al.*, 2012). Diaz and Jones (2004) uses the temporal profiles of queries to predict query performance. They divide queries into three categories according to their temporal profiles: temporally unambiguous, temporally ambiguous, and atemporal. They use a classification algorithm to predict the performance of the queries using features such as the temporal *KL-divergence*, autocorrelation, kurtosis, and other burst-related features. Kulkarni *et al.* (2011) proposed a temporal query categorization based on two dimensions: popularity changes, and content changes. In terms of popularity they look into four characteristics: the number of spikes, the shape of the spikes, query periodicity, and the overall trend. Content changes are measured using tf-idf as a query-dependent measure and the dice coefficient as a query-independent metric.

The Temporalia track at NTCIR (Joho *et al.*, 2014) has a temporal query intent categorization (TQIC) task that proposed classifying queries into four sub-classes: *past*, *recency*, *future*, and *atemporal queries*. Gupta and Berberich (2015) proposed to use multiple levels of granularity to solve the problem of temporal query classification. At the top level they consider two classes: *temporal* and *atemporal*. Under temporal queries they consider two classes: *temporal unambiguous* and *temporal ambiguous*. Temporal ambiguous queries are divided into three classes: *year*, *month*, *day*. The year class further subdivides into two classes: *periodic*, and *aperiodic*. Their approach is similar to the work of Kanhabua and Nørvåg (2010) in the sense that it leverages on temporal

expressions and publication dates in pseudo-relevance documents to classify implicit temporal queries. The classifier achieved a good accuracy in temporal query classification, however *temporally unambiguous* and *aperiodic* queries are harder to classify by looking only at pseudo-relevant documents.

There are two main types of time-sensitive queries:

- Temporal search patterns
- No temporal search patterns but relevance is time-dependent

Often temporal queries exhibit temporal patterns either in query logs or in the timeline of the results (Jones and Diaz, 2007). However, Cheng *et al.* (2013) studied "timely queries", a class of queries that favor recent documents, which have no major spikes in either document of query volumes over time.

2.2.6 Time-Aware Query Auto-Completion and Diversity

Query auto-completion is a common feature in modern search engines that can help users quickly define in more detail the information need by selecting auto-completion suggestions in real-time as they type. Typically, query auto-completion methods predict the most likely completions using only information about the most popular queries in the past using query logs as the main source of information. However, query popularity changes over time and the ranking of completions must take into account the time dimension. That is, suggestions should reflect the changes in popularity during the course of the year to account for the changing interests of the users.

Shokouhi (2011) proposed using long-term time-series decomposition for detecting seasonal queries such as annual events and recurring events. Shokouhi and Radinsky (2012) extend this work using time-series modeling to rank candidates according their forecasted frequencies to obtain more reliable suggestions. Modeling the temporal trends of queries can significantly improve the ranking of query auto-completion candidates. Predictions based on aggregated frequency over shorter more recent time periods was found to perform better than using long-term data.

Cai *et al.* (2014) studied how to combine time-sensitivity and user personalization for query auto-completion. They propose first using auto-correlation to detect query seasonality by long-term time-series analysis, and using a regression model to predict query popularity trends. Only then, the top query formulation candidates are re-ranked by taking into account the similarities with the user's past search behavior, including past search sessions. On the other hand, Whiting and Jose (2014) proposed predicting query popularity using only recently observed query popularity trends. Their results showed that for predictions after 2 or 3 keystrokes this approach can outperform a non-temporal query auto-completion baseline. Styskin *et al.* (2011) proposed a machine learning approach to identify recency-sensitive queries. They propose diversifying search results by promoting results according to the predicted recency level.

For temporally ambiguous queries identifying the relevant time periods is hard. Berberich and Bedathur (2013) proposed to simply focus on diversifying the time periods covered by the retrieval results for temporally ambiguous queries. Whiting *et al.* (2013) studied the temporal variance of query intents for event-driven queries on Wikipedia. They argued that event-driven queries have highly temporally variable subtopics, or query intents. Using query logs they found that many event-driven queries can be mapped to article sections. They also found that popularity changes in query intents in Wikipedia is correlated to the rate of Wikipedia article sections edits.

Zhou *et al.* (2013) studied the popularity changes of pages contained in Wikipedia's disambiguation pages. They found that traditional diversity metrics for ambiguous queries are impacted negatively when the subtopic popularity is considered static over time. Nguyen and Kanhabua (2014) leverage on multiple sources of information (i.e., query logs and the collection) to develop time-aware diversification methods. They proposed diversifying search results for temporally ambiguous or multi-faceted queries by first identifying temporal subtopics and taking recency into account, which outperformed the baselines on these query categories.

2.3 Pseudo-Relevance Feedback

Pseudo-relevance feedback (PRF) is an automatic query expansion technique, which was shown to significantly improve results in microblog retrieval (Fan *et al.*, 2015; Lin and Efron, 2013b; Lin *et al.*, 2014). In a comparative study conducted by Zhai and Lafferty (2001), RM3, a variant of the relevance model, was found to be one of the most effective pseudo-relevance feedback methods. Carpineto and Romano (2012) claim that Pseudo-relevance feedback is not yet a standard feature in most production search engines and attribute this fact to the efficiency issues raised by pseudo-relevance feedback at retrieval time. Most production search engines have strict query response time requirements, since long response times are correlated with query abandonment.

The standard implementation of the pseudo-relevance feedback algorithm involves a two-retrieval process:

- initial retrieval of feedback documents to generate the expanded query
- re-retrieval using the final expanded query

In pseudo-relevance feedback the top-ranked documents retrieved, denoted by $\mathcal{R}$, using a standard retrieval model from an index of the search collection are used for expansion. For term selection, relevance models by Lavrenko and Croft (2001) provide a framework for estimating the probability distribution, θ_F, over possible query terms, w, given an initial query, q, according to the equation

$$P(w \mid \theta_F) \propto \int_d P(w \mid d) \cdot P(q \mid d) \cdot P(d) \tag{2.7}$$

where d is a document language model and $P(q \mid d)$ is the query-likelihood. To make the calculation of the relevance models feasible, Lavrenko and Croft (2001) suggest approximating the integral by a summation over language models of the

top ranked documents, $\mathcal{R}$, which gives us the following query model estimate:

$$P(w \mid \theta_F) \propto \sum_{d \in \mathcal{R}} P(w \mid d) \cdot P(q \mid d) \cdot P(d) \tag{2.8}$$

$$\propto \frac{1}{|\mathcal{R}|} \sum_{d \in \mathcal{R}} P(w \mid d) \cdot P(q \mid d), \tag{2.9}$$

for every term w in the vocabulary, where the last step follows from considering a uniform contribution of the language model of each top ranked document retrieved, $P(d) = \frac{1}{|\mathcal{R}|}$. The relevance model $P(w \mid \mathcal{R}) \approx P(w \mid \theta_F)$ for query q is a weighted average of the terms in the top documents retrieved, where the weights are the query-likelihood scores for the query q.

2.3.1 Time-Based Pseudo-Relevance Feedback

A few time-based pseudo-relevance feedback methods were proposed for retrieval in time-sensitive collections using the relevance model framework. Amodeo *et al.* (2011) analyzed the relationship between time and relevance on the Blogo6 collection. They found that the publication dates of retrieved documents are close together in time and loosely matches the distribution of relevant documents. They exploited this correlation to improve retrieval effectiveness employing a Rocchio-like time-based query expansion algorithm. Keikha *et al.* (2011) proposed time-based relevance models where they assume that the publishing date has an effect on the terms. They introduce a generative model of the query that first selects a date and then a term based on the time and query:

$$P(w \mid \theta_F) = \sum_{T} P(w \mid T, q) \cdot P(T \mid q) \tag{2.10}$$

$$\propto \sum_{T} \frac{1}{|\mathcal{R}_T|} \sum_{d \in \mathcal{R}_T} P(w \mid d) \cdot \frac{\sum_{d \in \mathcal{R}_T} P(q \mid d)}{\sum_{T'} \sum_{d \in \mathcal{R}_{T'}} P(q \mid d)} \tag{2.11}$$

where $P(w \mid T, q)$ is the importance of the word w in day T for the query q and $P(T \mid q)$ is the importance of day T to a query q, which can be estimated over temporal slices of pseudo-relevant documents, $\mathcal{R}_T$.

They found this approach was able to improve the coverage of the expanded query over the different subtopics by using temporal information to weights and select expansion terms. Experiments on the Blogo8 collection showed that this method can outperform standard relevance models estimation.

Choi and Croft (2012) extend the framework proposed by Keikha *et al.* (2011) by making a simplifying assumption that $P(d \mid T, q)$ can be equal to $P(d \mid q)$ since the temporal dimension is incorporated already in choosing d. Therefore, their final equation for the new simplified formulation to compute a time-based relevance model:

$$P(w \mid \theta_F) \propto \sum_T P(T \mid q) \frac{1}{|\mathcal{R}_T|} \sum_{d \in \mathcal{R}_T} P(w \mid d) \cdot P(q \mid d). \tag{2.12}$$

In this formulation, a relevance model for each time T is estimated using the retrieved documents published in time T, $\mathcal{R}_T$. Each of the relevance models are then weighted by some estimate of $P(T \mid q)$ to get the final expansion terms over all the time periods.

Miyanishi *et al.* (2013) introduce a similar temporal pseudo-relevance feedback approach relying mainly on Dakka *et al.* (2012) to derive an analogous relevance model that integrates temporal information:

$$P(w \mid \theta_F) = \sum_{d \in \mathcal{R}} P(w, d \mid q) \tag{2.13}$$

$$= \sum_{d \in \mathcal{R}} P(w, w_d, t_d \mid q) \tag{2.14}$$

$$= \sum_{d \in \mathcal{R}} P(w, w_d \mid t_d, q) \cdot P(t_d \mid q). \tag{2.15}$$

Following Efron and Golovchinsky (2011) and making a simplifying assumption that $P(w, w_d \mid t_d, q)$ can be equal to $P(w, w_d \mid q)$ since the temporal relevance, t_d, is independent of the document content w_d.

$$P(w \mid \theta_F) \propto \frac{1}{\mathcal{R}} \sum_{d \in \mathcal{R}} P(w \mid w_d) \cdot P(w_d \mid q) \cdot P(t_d \mid q). \tag{2.16}$$

Metzler *et al.* (2012) defined *microblog event retrieval* as a search task that goes beyond *ad hoc* retrieval. Given a query that describes an event, the goal is to retrospectively retrieve from microblog archives a ranked list of structured event representations. The structured event representations consists in a summary of the relevant timespans when an event occurred and was actively discussed. To uncover these subtopics, from microblog streams of very short and noisy posts, they proposed a temporal query expansion technique combining pseudo-relevance feedback with term burstiness, and other temporal features. Their technique divides the timeline into timespans of 1 hour, and ranks them according to the proportion of messages posted during the timestamp that match the query terms. Then a burstiness score is calculated for the terms in each timestamp, by calculating the ratio between the term's likelihood on the timespan versus the likelihood of it occurring during any of the timespans. This is used to weight terms, so that when counts of term occurrences are higher than usual the term's will have a higher weight in the final model. To improve the robustness of the model, they proposed the use of the geometric mean to combine the burstiness scores of multiple timespans so that a single timespan is not able to cause query-drift.

Whiting *et al.* (2012) proposed an approach to pseudo-relevance feedback based on the n-grams extracted from the top tweets in the initial retrieval, and taking into account their temporal profile. They combine pseudo-relevant document term distribution and temporal collection evidence using a variant of PageRank (Page *et al.*, 1999) over a weighted graph that models the temporal correlation between n-grams. Peetz *et al.* (2013) proposed to leverage instead on the temporal distribution of the pseudo-relevant documents themselves. For each query, *bursty* time periods are identified and documents from these periods are then selected for feedback. The query model is updated with additional terms from the documents found in the time periods selected, which are assumed to be of higher quality. Following the same rationale, that term expansions should be biased to draw from documents from relevant (bursty) time periods Rao and Lin (2016) proposed capturing these by estimating the parameters of a continuous hidden Markov model that best explains the sequential dependencies

in the temporal distribution of documents retrieved in the initial feedback step, computing the most likely state sequence using the Viterbi algorithm, and drawing terms only from bursty states. This approach yielded better results than relevance models calculated using the RM3 method and KDE variants.

2.3.2 Leveraging External Collections for PRF

A single large external corpus that is more reliable and possibly less noisy than the target collection can sometimes be used to improve the effectiveness of query expansion. Likewise, when the target retrieval collection is too large to be used for feedback without raising efficiency concerns relevance models can be estimated using an external corpus only.

Arguello *et al.* (2008) and Elsas *et al.* (2008) proposed using Wikipedia as the external query expansion corpus to estimate relevance models for blog feed recommendation and search. They found that using Wikipedia to estimate the relevance models outperformed the retrieval effectiveness of relevance models build with the target collection.

Xu *et al.* (2009) proposed using external sources to obtain extra lexical information for pseudo-relevance feedback. They estimate relevance models using an external Wikipedia corpus for query expansion. Furthermore, their approach is query-dependent and can categorize queries into three types based on information from Wikipedia outperforming the relevance model baseline. While the authors focused on Wikipedia as the single source for pseudo-relevance feedback, they argue that combining the target collection and Wikipedia as sources could avoid some instances of degraded performance from using either source separately.

Bendersky *et al.* (2012) argue that standard query formulation tasks such as term weighting and query expansion often use a single source of information. They argue for employing multiple information sources in information retrieval tasks. In their query expansion experiments, they combined multiple information sources from newswire and web corpora and found better retrieval effectiveness than with a single source.

Weerkamp *et al.* (2012) developed a novel query modeling framework to combine evidences from multiple external collections. As a starting point they take the usual route by modeling the query as a linear combination of the original query model $P(w \mid q)$ and the expanded query model $P(w \mid \theta_F)$:

$$P(w \mid \theta_q) = \lambda \cdot P(w \mid q) + (1 - \lambda) \cdot P(w \mid \theta_F) \tag{2.17}$$

$$= \lambda \cdot \frac{\#(w, q)}{|q|} + (1 - \lambda) \cdot P(w \mid \theta_F) \tag{2.18}$$

They proposed to estimate the expanded query model by a mixture of a number of collection-specific query expansion models. Given $\mathcal{C}$ a set of external collections used for query expansion they combine evidence from multiple external collections to reach the final model to estimate the probability of a term w in the expanded query θ_F.

The finalized instance of the proposed External Expansion Model (EEM) is

$$P(w \mid \theta_F) \propto \sum_{c \in \mathcal{C}} P(q \mid c) \cdot P(c) \sum_{d \in c} P(w \mid d) \cdot P(q \mid d) \cdot P(d \mid c), \tag{2.19}$$

it accounts for the prior probability of a collection $P(c)$, the query-dependent collection importance $P(q \mid c)$, the term probability $P(w \mid d)$, the document relevance $P(q \mid d)$, and the importance of a document in a given collection $P(d \mid c)$. An interesting property is that, if we assume that the query-dependent collection importance $P(q \mid c)$ is uniformly distributed and that the importance of a document in the collection $P(d \mid c) = \frac{1}{|\mathcal{R}_c|}$, we arrive at the formulation of Mixture of Relevance Models (MoRM) proposed by Diaz and Metzler (2006).

$$P(w \mid \theta_F) \propto \sum_{c \in \mathcal{C}} \frac{P(c)}{|\mathcal{R}_c|} \sum_{d \in \mathcal{R}_c} P(w \mid d) \cdot P(q \mid d) \tag{2.20}$$

Mixture of Relevance Models (MoRM) proposed by Diaz and Metzler (2006) used larger auxiliary external corpora to improve the estimation of relevance models. They built enhanced relevance models by combining a relevance model estimated using the target collection and an additional relevance model

estimated using a large external corpus. This approach was shown to improve the estimation of relevance models leading to improved retrieval effectiveness.

2.3.3 Efficiency Constraints of PRF

The evaluation of Relevance Models (RM) is very costly. Cartright *et al.* (2010) and Lavrenko and Allan (2006) identified two major inefficiency factors: the number of terms in the relevance model, and the number of documents in the index. Traditionally, to speed up the evaluation of relevance models, the number of expansion terms is pruned to reduce the size of the final query, because the number of terms in the relevance model is one of the major inefficiency factors (Lavrenko and Allan, 2006; Metzler *et al.*, 2005).

Lavrenko and Allan (2006) proposed shifting most of the computational cost from retrieval time to indexing time to improve the efficiency of relevance models. At indexing time, they compute a similarity index from a document similarity matrix between all documents. At retrieval time, they lookup the similarity index to find additional relevant documents. However, the computational cost of precomputing the similarity index might be prohibitive for large corpora. Similarly, Wurzer *et al.* (2016) proposed pruning the number of documents ranked in the final retrieval step using Locality Sensitive Hashing (LSH). At indexing time, they use a standard LSH scheme to assign hash-codes to each document in the collection based on their position within the vector space. At retrieval time, they generate a relevance model and assign it a hash-code. Matching this hash-code against the collection yields a bucket of candidate documents. This bucket links to a smaller number of candidate documents in the index that are ranked in the final retrieval step, instead of issuing an expensive re-retrieval on the complete index, improving response times.

Traditionally, Pseudo-relevance feedback is implemented using two retrieval steps, an initial retrieval and a re-retrieval using the final expanded query. Diaz (2015) has recently shown that a final re-retrieval might not be necessary to achieve a good effectiveness in traditional TREC corpora. Condensed List Relevance Models (CLRM) (Diaz, 2015), replaces the computationally expensive

re-retrieval with the re-ranking of the original feedback documents only. This implementation could produce near identical effectiveness, while avoiding entirely the cost of evaluating the relevance model over the total number of documents.

2.4 Resource Selection

Very large document collections are often partitioned into various shards for indexing; processing is distributed across multiple nodes and indexing time is reduced. Query processing can be performed in parallel across multiple nodes, which improves search time. Nonetheless, to reduce the number of shards involved in each query, it is critical to find a good allocation policy, since the number of shards is dictated by the document-to-shard allocation policy.

There are simple allocation policies such as random and source-based and others more advanced such as topic-based. Kulkarni and Callan (2010) study the trade-offs between cost and accuracy of using topic-based shards for selective searching just a few machines in a distributed index environment. They found that search cost was reduced to less than 1/5th without impacting accuracy across three large collections. In a federated search system, pruning the number of documents ranked is the major reason for the reduction of the total query-evaluation workload. At index time, the collection is partitioned into several smaller topical index shards (or verticals). At retrieval time a resource selection algorithm selects the most likely shards for retrieval, which improves efficiency.

Ogilvie and Callan (2001b) proposed generating expanded queries using only the central sample index for feedback, an index containing a representative sample of all the collections, and then submitting the same query to all the selected search engines in a "one query fits all" approach. In contrast, Shokouhi *et al.* (2009) proposed a "query-specialization" approach, by generating a *local* expanded query using a sample of each collection separately and submitting *focused* queries to each search engine to avoid topic drift or vocabulary mismatch. These works provided evidence that query expansion can be applied effectively in a federated information retrieval system.

A critical element in this framework is the selection of the resources to be searched. There are two main types of resource selection algorithms: 1) sample-document methods, and 2) vocabulary-based methods.

2.4.1 Sample-Based Methods

Sample-document methods rely on a *centralized sample index* (CSI) built by combining into a single index representation sets of all collections. The representation set for each collection are usually built by sampling its documents anywhere from 1% up to 10%. Therefore, the size of the resulting *centralized sample index* is the main factor for a more efficient resource selection method.

Previous studies have shown that sample-document approaches are highly effective. In sample-document resource selection algorithms, the user's query is run against the *centralized sample index* and the top-k retrieved documents are used in a voting fashion to select the final collections ranking. ReDDE, one of the first sample-document methods proposed by Si and Callan (2003) has the following general procedure:

- first, an initial retrieval list is obtained from the *centralized sample index* by retrieving the query using the query-likelihood retrieval model;
- second, the resource selection algorithm takes into account the ranking and/or scores of the documents and their provenience to calculate a ranking of the most likely collections;
- finally, the number of selected collections is truncated so that only the most likely collections are selected for searching.

CRCS (Shokouhi, 2007) *Central-Rank-based Collection Selection* (CRCS) method, uses a similar approach and scores resource i using:

$$\theta_i = \frac{1}{C_{max}} \cdot \frac{C_i}{S_i} \sum_{d \in S_i} I(d_j), \tag{2.21}$$

where C_i is the number of documents in collection i and S_i is the size of its representation set, the number of documents in the CSI sampled from collection i. The CRCS(exp) scoring variant, assigns a score to each document based on its ranking position using the following negative exponential impact factor:

$$I(d_j) = \alpha \, \mathrm{e}^{-\beta \cdot j}, \tag{2.22}$$

where j is the rank of document d and α and β are two constants tuned to 1.2 and 2.8, respectively. The shard sizes C_i are normalized using the size of the largest collection involved (C_{max}). The number of resources search is typically parameterized, and CRCS searches a static limit k of resources for all queries (sources or verticals).

Rank-S (Kulkarni *et al.*, 2012) contains a mechanism to dynamically limit the number of resources searched for each query. Considering n_i the number of documents in the top-k mapped to the sample S_i, a score for the collection C_i is computed as:

$$\theta_i = \frac{C_i}{S_i} \cdot n_i, \tag{2.23}$$

where C_i is the number of documents in collection i and S_i is the size of its representation set. Normalizing scores θ_i to get a probability distribution $P(i)$.

2.4.2 Vocabulary-Based Methods

Vocabulary-based resource selection algorithms, such as Taily (Aly *et al.*, 2013), represent each collection by its vocabulary statistics only. Kanoulas *et al.* (2010) observed that retrieval language modeling scores are gamma distributed. They used the *tail* of the distribution to estimate a cutoff score for relevant documents. Aly *et al.* (2013) proposed **Taily**, a vocabulary-based resource selection algorithm, records the mean and variance of each term's score across collections to be able to make this prediction. Taily, leverages on this information to predict the potential number of highly relevant documents contained in each collection and use this to rank collections.

2.5 Summary

Lots of research point to the benefits of addressing the temporal dimension in information retrieval tasks. In recent years, a lot of the research springing from academia has been finding their way into production Web search engines and search products on social media. In the industry, Web search engines have particularly benefited from using temporal aspects in their implementations of query auto-completion to predict the most likely users' queries. In fact modeling the temporal evidence in query logs for prediction query completions can solve a lot of issues. For instance, for the prefix *ha* Web search engines now suggest *harry potter* only when out of the *halloween* season when the latter is predicted. Modeling query seasonality was crucial to allow the learned model to better predict the completions of queries given the initial keystrokes of the query and their performance on a homologous period.

Since people crave for fresh new information, the official Twitter search engine used to display the results sorted in reverse chronological order and disregarding the degree of lexical match. This approach has served them well since the lexical retrieval score in such short texts is often not enough to provide the best ranking. They have since moved to a machine learning algorithm to calculate and rank their top results that appears to boost documents that received a lot of social activity (retweets and replies), and inter-speed them with an assortment of fresh matching results.

However, there appears to be no system in-place to help find relevant documents that do not necessarily match the terms used in the query. Query expansion is then, I feel, one of the most fundamental features that remain to be implemented in the search engines of social media networks. A social media search engine's query expansion implementation shares some problems with query auto-completion systems. Query expansion should take into account trends and freshness for the best performance. Most of the previous work has been focused on the temporal evidence obtained from a single main source of information. In related work, this has been typically the corpus itself or large query logs from production search engines.

VERTICAL PSEUDO-RELEVANCE FEEDBACK ARCHITECTURE FOR MICROBLOG SEARCH

3

"Efficiency is doing things right;
Effectiveness is doing the right things."
— Peter Drucker

Users search on Twitter to find and monitor the most up-to-date information on current events and people of interest, often using short under-specified keyword queries as new information is arriving at a high-rate. Teevan *et al.* (2011) found that queries submitted to Twitter were significantly shorter than queries submitted to Web search engines. Moreover, since the target documents are short, there is also a higher mismatch between the keywords users employ in their searches to specify the information need and relevant documents, which is known as the *vocabulary mismatch problem* (Furnas *et al.*, 1987).

Query expansion (QE) can provide a richer representation of the information need and are therefore essential in microblogs. Query expansion is often used to increase recall by matching more documents. However, it can also produce better document rankings leading to increased retrieval effectiveness. For this purpose, several automatic query expansion methods leverage on external static data such as dictionaries, domain-specific thesauri or precomputed corpus-specific information. In microblogs, query expansion should be based on information that has good coverage of real-world events in which searchers are interested.

In standard pseudo-relevance feedback (PRF) the collection itself is used for feedback. Therefore, PRF-based query expansion methods can and have been widely used to improve search in many diverse collections. It was shown to be essential in previous microblog search approaches (Lin and Efron, 2013b; Soboroff *et al.*, 2012). In standard PRF the top-k documents retrieved by the initial query (feedback documents) are assumed to be relevant, which avoids the need for users' relevance feedback. Term selection and weighting can be

supported by terms extracted from feedback documents and collection statistics using relevance models (RM) (Lavrenko and Croft, 2001).

In order to compute the expansion terms for a query using PRF, it is then necessary to issue an initial query to a search engine over the whole collection, which does not scale well to large volumes of data. In most production retrieval systems, caching of posting lists and search results significantly reduces the workload of back-end servers, especially for popular queries, and provides shorter average response times (Baeza-Yates and Ribeiro-Neto, 1999).

However, in fast-moving collections (e.g., newswire, microblog) lots of documents are being added to the index continuously. Thus, the top results for a query can change quickly over time. The ephemeral nature of information seeking in microblog search calls for an architecture that provides fresh expansion terms to obtain better retrieval results. Therefore, while caching could provide a more efficient PRF process, it might be prone to delays that could lead to outdated query expansions and poor user experience for microblog search.

Query expansion can be essential in microblog search to obtain good search results due to the short size of queries and documents. Since information in microblogs is highly dynamic, pseudo-relevance feedback (PRF) on an up-to-date index increases the chances of retrieving relevant documents. In this chapter, we focus on the research question: *can we deliver PRF of comparable effectiveness at a lower retrieval cost?*

This study focused on the following questions: *how can we deliver query expansion with comparable effectiveness to standard PRF at a lower computational cost?*, and more specifically, *how to achieve it in scenarios where information needs are highly dynamic?* The proposed solution brings two major advantages over standard query expansion approaches in microblogs:

- Reduced computational cost of query expansion by leveraging on an external corpus of Twitter news sources instead of the whole search collection. It departs from previous works that mainly use a single information stream and moves towards an architecture where multiple information streams are continuously feeding the different query expansion corpus.

- Federated search approach (Shokouhi and Si, 2011) to the query expansion process, leveraging on resource selection algorithms to select *query-specific* news verticals. The novel vertical feedback design is crucial to unlock the efficiency potential of the proposed query expansion architecture. In this chapter, query-dependent query expansion is achieved by employing a federated search architecture over a set of vertical indexes for pseudo-relevance feedback.

The proposed Pseudo-Relevant Vertical Feedback (PRVF) architecture reduces the work-load of the whole search engine for query expansion because it selects a few news verticals and only those are then searched to obtain feedback documents for query expansion. The rationale is that with effective resource selection algorithms it may outperform standard PRF.

3.1 Federated Query Expansion Architecture

In the PRF approach, the cost of the query expansion process is tied to the cost of the initial retrieval. Alternatives include the use of external data, such as dictionaries or other static external corpora (e.g., Wikipedia) where the costs of generating an expanded query can be smaller. Especially in large collections, the use of PRF introduces a heavy computational demand. Tackling this major challenge requires more efficient query expansion techniques that can reduce the initial retrieval cost and at the same time deliver a high-quality query expansion process that provides retrieval effectiveness comparable to standard techniques.

Note that in the relevance model approach to PRF, the term selection step has a relatively low computational cost, since it uses data that is readily available after the initial retrieval, with a *query-likelihood* retrieval model, such as the scores and term vectors of the top-k documents. In large text document collections, term vectors can be stored during indexing and fetched when needed for a space-time trade-off. In short text document collections, term vectors can be computed on-the-fly rapidly since documents are short.

Real-time microblog search presents an additional challenge for query expansion, related to the velocity with which the searchers' interest changes, following shifts in trending topics on social media and on the news. Although Wikipedia has been used as an alternative query expansion corpus for blog search (Arguello *et al.*, 2008) with significant improvements in effectiveness, it would not be a good fit for real-time microblog retrieval since it would not be able to provide the most up-to-date expansion terms. Thus, in light of our goal, a more dynamic, reliable, and concise external corpus must be considered.

The proposed approach leverages on an external news corpus that is partitioned into news verticals for efficient pseudo-relevance feedback in real-time microblog search. The news corpus is highly dynamic and is maintained up-to-date as new documents are arriving to be indexed. This novel federated query expansion architecture stems from a new understanding of how temporal and topical information is searched in microblogs. The fundamental properties of our contribution are:

- **Efficient query expansion.** How much (and what type of) data is required to get query expansion that is more efficient in microblog search.
- **News sources for query expansion.** The use of news posts from Twitter covers the information seeking behavior of users in the microblog search scenario. Current events are reported live as they unfold by online news sources.
- **News verticals for query expansion.** We propose an organization of news sources into topic-based verticals. A federated search approach based on verticals affords extra cost reduction because resource selection algorithms can be used to select only a few *query-specific* verticals.

The question then arises, *"How to partition the index of news documents into topical index shards?"*. This choice may influence the latency and efficiency of the query expansion process. Previous research found that topic-based shards offer the best balance of retrieval effectiveness and efficiency (query processing costs) (Hafizoglu *et al.*, 2017; Kulkarni *et al.*, 2012; Larkey *et al.*, 2000).

While with uniform sharding the work is distributed across all machines, so that it can be done more quickly, it does not reduce the total work done since all the shards are involved for every search query. When a CSI or *broker* is used to select the most useful verticals for each query, only those verticals are searched, instead of all of them. This approach reduces the amount of work done for each query, and since the selected resources can be searched in parallel, it can also be much faster. In a *cooperative* (Shokouhi and Si, 2011) federated search environment global corpus statistics can be accessed by each federated search engine or by a *broker* and therefore merging results from multiple sources or verticals is straightforward.

3.1.1 Computational Cost of PRF

The standard PRF procedure creates an expanded query with new terms extracted from the top-k documents in the initial ranking obtained by retrieving with the original query terms. Firstly, the user's query q is issued to the system to retrieve a ranked list of documents $\mathcal{R}(q, D)$, over the whole corpus D using a *query-likelihood* (QL) retrieval model. Secondly, the top-k documents retrieved, $\mathcal{R}(q, D)$, are used to build an expansion language model θ_F with the terms extracted from those documents. The final ranking is obtained by issuing the final query $\theta_{q'}$, which is a linear model combination of the original query language model θ_q and the expansion language model θ_F with parameter λ, as follows:

$$\theta_{q'} = (1 - \lambda)\theta_q + \lambda\theta_F. \tag{3.1}$$

The PRF cost can be expressed as the sum of two components: $C_{QE}(q, D)$, the cost of retrieving the ranked list of pseudo-relevant documents using query q, and $C_R(q', D)$, the cost of retrieving the final documents using the final expanded query q'. Formally, the pseudo-relevance feedback cost is defined as

$$C_{PRF}(q, D) = C_{QE}(q, D) + C_R(q', D), \tag{3.2}$$

where the cost metric C_R is based on previous work showing that the sum of the

lengths of the posting lists that need to be accessed for each query is strongly correlated with query response times Macdonald *et al.*, 2012; Moffat *et al.*, 2007. Hence, for standard PRF, $C_{QE}(q, D) = C_R(q, D)$, and following Aly *et al.*, 2013; Kulkarni and Callan, 2015; Kulkarni *et al.*, 2012; Macdonald *et al.*, 2012; Moffat *et al.*, 2007, we define C_R.

Definition 1 (Single-step retrieval cost) *The cost of retrieval for a given query q is calculated as*

$$C_R(q, D) = \sum_{q_1 \cdots q_n} postings_D(q_j), \tag{3.3}$$

where $postings_D(w)$ is the number of accessed postings for term t.

In the relevance model approach to PRF, the term selection step is constant for all PRF methods when we consider a fixed top-k number of documents. Hence, we discard this part of the cost because we are interested in relative costs. Previously designed optimization techniques can lead to better efficiency and cost savings in the PRF retrieval process Cartright *et al.*, 2010; Lavrenko and Allan, 2006. However, these involve precomputations and would not be a good fit for dynamic collections.

Since dynamic pruning techniques were not the focus of this study, the metrics used reflect the cost of the full evaluation of queries, using the full postings lists. Dynamic pruning strategies such as Weighted AND (WAND) (Broder *et al.*, 2003) and Block-Max WAND (Petri *et al.*, 2013), can optimize the evaluation of queries by pruning the scoring of documents that cannot make the final top-k. Kim *et al.* (2016) found that the efficiency improvements from WAND are larger on selected shards (topical index partitions). Therefore, dynamic pruning can improve efficiency further.

3.1.2 Expansion with External Corpus

The use of external corpora for query expansion has been studied in fields as far apart as blog search Arguello *et al.*, 2008 and Web search Diaz and Metzler, 2006. Arguello *et al.* (2008) addressed blog search with Wikipedia

as an alternative query expansion corpus with significant improvements in effectiveness. Wikipedia and other external sources have been used in Microblog search offering a wide coverage for past events Lin and Efron, 2013b; Qiang *et al.*, 2015. In light of our goal, we aim to use *the most up-to-date, reliable, and concise external corpus*.

To create a Microblog expansion corpus, there are several possible strategies. Sampling posts from Twitter accounts picked at random can lead to a low quality expansion collection. Instead, using multiple authoritative news sources lends redundancy to the system since multiple news sources often report the same news story. Hence, since many news outlets use Twitter for the dissemination of news articles, we propose to listen to the stream of news headlines directly from their Twitter profile pages (i.e., *timelines*). This corpus can be orders of magnitude smaller than the retrieval corpus.

The implemented approach relies on an external news corpus covering multiple authorative sources for expanding the query (we used 70 news sources). The news corpus is highly dynamic and is maintained up-to-date as new documents are arriving to be indexed – current events are reported live as they unfold by online news sources. As a consequence of this dynamic environment, the *query expansion corpus age and time span* will play a major role in the quality of the expansion corpus.

The use of news sampled from Twitter covers the information seeking behavior of users in the microblog search scenario. This phenomenon is nicely tied to the natural topical bias of each query, suggesting that partitioning the expansion corpus into news verticals will bring greater benefits in terms of precision and expansion cost.

3.1.3 PRVF: Pseudo-Relevant Vertical Feedback

The federated query expansion architecture stems from a new understanding of how temporal and topical information is searched in microblogs. To take full advantage of the external expansion corpus the organization of documents into index shards is fundamental. This architectural decision influences the latency

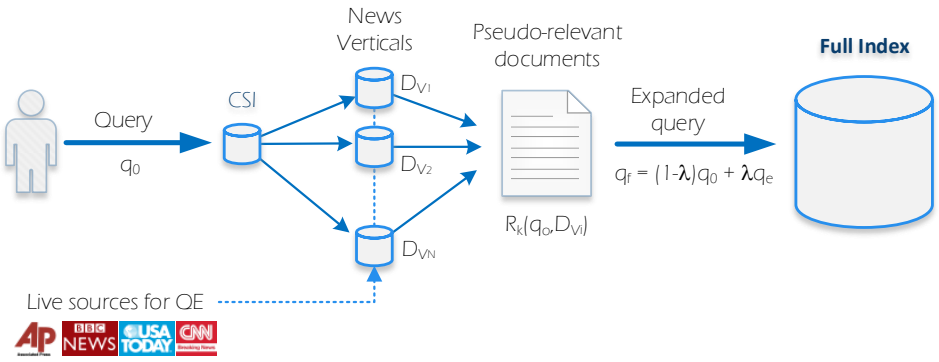


Figure 3.1: PRVF– Pseudo-Relevant Vertical Feedback.

and efficiency of the query expansion process. Uniform sharding distributes the work across all machines so that it can be done more quickly, but it does not reduce the total work done and all shards are involved in every query. Previous work found that topic-based shards offer the best balance between effectiveness and efficiency (Hafizoglu *et al.*, 2017; Kulkarni *et al.*, 2012). Hence, we organize sources into topic-based verticals, see Table 3.1.

Queries are routed through a *broker* to a subset of the most useful verticals. To make this decision, the broker keeps a *central sample index* (CSI) of all verticals to select the few verticals to search for each query. This approach reduces the amount of work done for each query and, since the selected verticals can be searched in parallel, it can be much faster. In a *cooperative* Shokouhi and Si, 2011 federated search environment, global corpus statistics can be accessed by each federated search engine and by the *broker* and therefore merging results is straightforward.

An illustration of the architecture is shown in Figure 3.1. It represents the use of a sample-document resource selection algorithm, where a *centralized sample index* (CSI) indexes a representation sample of each vertical's documents. *CSI* denotes the collection containing a random sample of collection S, which is used as CSI. The top-k documents from collection CSI $\mathcal{R}(q, CSI)$, in response to the initial query q, using a *query-likelihood* retrieval model, are processed by a resource selection algorithm.

Pseudo-Relevant Vertical Feedback (PRVF) is a query expansion architecture

that uses an external corpus organized into verticals to efficiently select expansion terms. In the proposed approach, the query expansion corpus is organized into a set of verticals $\mathscr{C} = \{c_1 c_2 \cdots c_{|\mathscr{C}|}\}$ from which a resource selection method selects the most likely set $\mathscr{C}_q$, which are then searched in parallel. Formally, we wish to compute

$$\mathscr{C}_q = \{c_1 c_2 \cdots c_{|\mathscr{C}_q|}\}, \tag{3.4}$$

a ranked set of $|\mathscr{C}_q|$ verticals selected given q, which are the most promising, in terms of relevance, from the full set of $|\mathscr{C}|$ verticals.

To select the verticals, a resource selection algorithm either (i) uses a *centralized sample index* (CSI) which indexes a representation sample *CSI* Kulkarni *et al.*, 2012; Shokouhi, 2007 of each vertical's documents, or (ii) uses the term statistics Aly *et al.*, 2013 of each vertical index. With CSI based algorithms, we retrieve $\mathscr{R}(q, CSI)$, the top-k documents from collection *CSI* in response to the initial query q, using the *query-likelihood* retrieval model. The verticals with more results in this sample are then selected for the feedback retrieval step. The key details of the implemented resource selection algorithms CRCS Shokouhi, 2007, Rank-S Kulkarni *et al.*, 2012 and Taily Aly *et al.*, 2013 are in Chapter 2.

Finally, the top-k documents retrieved from the verticals selected $\mathscr{C}_q$ are used to build the expansion language model q_F, hence *vertical feedback*.

$$\mathscr{R}(q, \mathscr{C}) = \bigcup_{i=1}^{|\mathscr{C}_q|} \mathscr{R}(q, c_i) \tag{3.5}$$

which is interpolated with the original query model q. This set of documents is then used to expand the original query, thus, ending the computation of q'.

3.1.3.1 Computational cost of PRVF

It is worth recalling equation 3.2, where we defined the cost of standard PRF. Now, with Pseudo-Relevant Vertical Feedback (PRVF), the query expansion cost C_{QE} is associated with C_{VF}, the cost of *vertical feedback*.

Formally, the PRVF cost can be defined using the following equation:

$$C_{PRVF}(q) = C_{VF}(q, \mathscr{C}) + C_R(q', D) \tag{3.6}$$

where C_{VF} is the cost of expanding q on a vertical architecture and C_R is the computational cost for searching the full index with the final query. The C_{VF} efficiency measure proposed in Aly *et al.* (2013) accounts for two separate costs:

$$C_{VF}(q, \mathscr{C}) = C_{SEL}(q, \mathscr{C}) + C_{VR}(q, \mathscr{C}) \tag{3.7}$$

where $C_{SEL}(q, \mathscr{C})$ is the cost of the resource selection algorithm, and $C_{VR}(q, c)$ (defined later) is the cost of retrieving documents in parallel from the selected verticals $\mathscr{C}_q$. The cost $C_{SEL}(q)$ depends on the type of algorithm used:

$$C_{SEL}(q) = \begin{cases} CSI(q) & \text{if sample-document} \\ |\mathscr{C}| & \text{if vocabulary-based} \end{cases} \tag{3.8}$$

where $|\mathscr{C}|$ is the total number of verticals and $CSI(q) = C_R(q, CSI)$ the number of postings accessed in the CSI for all the query terms in q considering a sample-document resource selection algorithm. In the vocabulary-based resource selection algorithms (Aly *et al.*, 2013), typically since a single look-up operation is performed, it is set to the total number of verticals $C_{SEL}(q) = |\mathscr{C}|$.

Definition 2 (Parallel retrieval cost) *In a vertical search scenario, C_R is calculated for a given query q using the number of postings accessed as follows:*

$$\begin{aligned} C_{VR}(q, c) &= \sum_{i=1}^{|\mathscr{C}_q|} C_R(q, c_i) \\ &= \sum_{i=1}^{|\mathscr{C}_q|} \sum_{q_1 \cdots q_n} postings_{c_i}(q_j) \end{aligned} \tag{3.9}$$

where $\mathscr{C}_q$ are the verticals selected by a resource selection algorithm and $C_R(q, c_i)$ is the number of postings in vertical i for all the terms in the initial query q.

3.1.3.2 Query response latency

A federated search architecture also affords faster response times via parallel work since multiple verticals can be searched in parallel. Considering the set of documents D, the latency metric C_{Lat} employed by Kulkarni and Callan (2015), quantifies the longest execution path C_L for a given query q, assuming a distributed query processing framework,

$$\begin{aligned} C_{Lat}(q,c) &= C_{SEL}(q,c) + C_L(q,c) \\ &= C_{SEL}(q,c) + \max_{i=1}^{|N|} \sum_{q_1 \cdots q_n} postings_{c_i}(q_j) \end{aligned} \tag{3.10}$$

where $postings_{c_i}(w)$ is the number of accessed postings in the inverted index for term w in vertical i, for a total of N verticals in a federated search environment.

3.1.4 Costs Comparison

The computational cost of the PRVF approach is strongly correlated to the cost of retrieving candidate documents in a federated search retrieval system. There might be non-negligible differences in the cost of the queries generated using different corpora for expansion. That said, if the number of terms in the expanded query is fixed for all methods, the cost of retrieving the final results $C_R(q')$ can be assumed to be of the same order of magnitude for all methods presented. Therefore, the first part of the cost equations Eq. (3.2) and Eq. (3.6), (i.e., the cost of the query expansion process, C_{QE}), can be used alone to compare both approaches in terms of computational cost:

$$C_{VF}(q,\mathscr{C}) << C_R(q,D) \tag{3.11}$$

$$\sum_{i=1}^{|\mathscr{C}_q|} \sum_{q_1 \cdots q_n} postings_{c_i}(q_j) << \sum_{q_1 \cdots q_n} postings(q_j) \tag{3.12}$$

The cost of the initial retrieval when using the whole collection is much larger than the proposed alternatives. Using an index built with the posts of

news sources provides a high-quality coverage of microblog user interests. Further computational gains are obtained by organizing news sources into topical verticals. The hypothesis is that *PRVF* offers the lowest query expansion computational cost, and can provide comparable retrieval effectiveness to standard PRF techniques that use the whole corpus for feedback. Experiments will now examine this hypothesis.

Table 3.1: Verticals and sources.

Vertical (c_i)	Accounts
general	abc, ap, bbcnews, bbcworld, cbsnews, cnn, cnni, foxnews, huffingtonpost, latimes, nprnews, nytimes, reuters, reutersuk, usatoday, mashable
politics	huffpostpol, politico, theeconomist, washingtonpost, wsj
technology	arstechnica, cnet, gizmodo, techcrunch, wired, wireduk, thenextweb, techrepublic, cnet, gigaom, macworld
sports	bbcsport, sinow, eurosport, eurosportuktv, sportscenter, espn
music	clash_music, rollingstone, nme, spinmagazine, stereogum, billboard, altpress, pitchfork
movies	americancine, thr, nytmovies, bbcfilms, totalfilm, guardianfilm, backstage, empiremagazine, filmcomment, timeoutfilm, sightsoundmag
entertainment	time, ew, variety, vanityfair, uncutmagazine
science	livescience, popsci, wiredscience, nasa, natgeo, newscientist
breaking	bbcbreaking, breakingnews, cnnbrk

Table 3.2: Number of relevant documents by source in the TREC datasets.

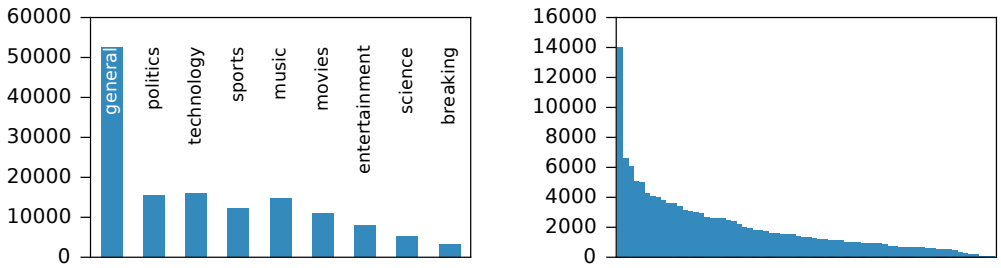
Source	2013	2014
AP	1	2
HuffingtonPost	3	0
Huff PostPol	1	1
BBCBreaking	0	1
BBCNews	0	1
BBCWorld	0	1
CNN	0	1
latimes	1	0
LiveScience	1	0
nprnews	0	1
NYTMovies	0	1
nytimes	0	1
politico	1	0
TIME	1	1
USATODAY	0	1
WiredUK	0	1
Total	9	13

3.2 Experimental Methodology

News sources were crawled from Twitter to create an external query expansion corpus. This external corpus of news was then partitioned into verticals thematically, as seen in Table 3.1. The proposed methods were evaluated on TREC Microblog datasets and compared strong retrieval baselines.

3.2.1 Microblog datasets

The Tweets2013 corpus was used with the topics from the 2013 and 2014 editions of the TREC Microblog track (Lin and Efron, 2013b). Tweets2013 is a microblog posts collection (approx. 240 million *tweets*) created by crawling Twitter's public sample stream over the period from 1 February 2013 to 31 March 2013. NIST provided relevance judgments on a three-point scale of "informativeness": not relevant, relevant and highly relevant.



(a) Documents per vertical. (b) Documents per source.

Figure 3.2: Number of documents in vertical-based and for source-based shards.

3.2.2 NewsSources corpus

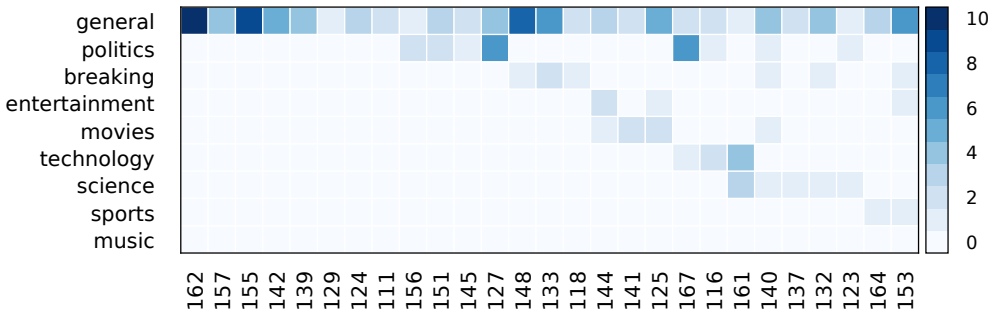
To build the news expansion corpus, 70 accounts from reliable news sources on Twitter, were selected. These accounts were carefully selected and belong to publications that are reputable and popular on Twitter.

Using the news headlines posted by the news sources accounts for query expansion might increase the chances of retrieving documents from these same accounts due to the bias introduced by using their vocabulary in the final query. To make sure that this bias does not improve the results unfairly, the intersection between the documents in the expansion corpus and the documents marked as relevant in the relevance judgments of the TREC datasets used was inspected. In total, only nine documents matched for TREC 2013 and for TREC 2014 there were only thirteen matching documents (see Table 3.2). Thus, there is no evidence to conclude that the choice of news sources affords an unfair advantage.

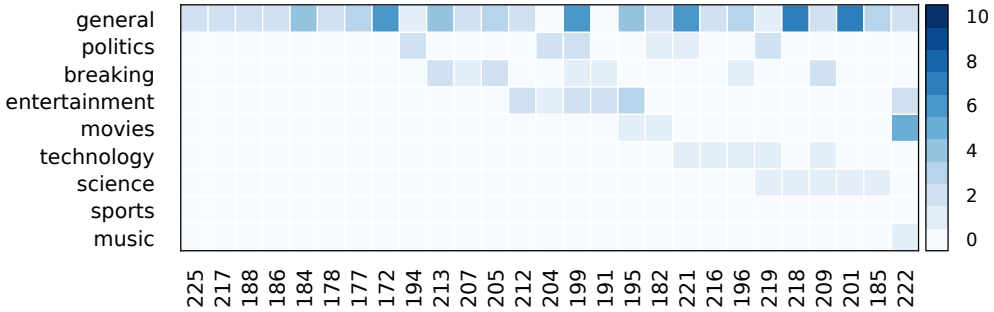
The *NewsSources* corpus (140,087 documents) contains the posts crawled from the timelines of the news sources on the same period covered by the target Tweets2013 corpus: 1 February 2013 – 31 March 2013 (inclusive). To use as a CSI For sample-document resource selection algorithms, a CSI is built using a smaller collection (16,687 documents). For practical reasons, the data used for the CSI was crawled from Twitter's ~1% and ~10% sample streaming APIs simultaneously. Documents from both streams were combined into a single index, which corresponds to a sample with almost ~11% of the size.

Informed by previous work on topical tweets clustering by Rosa *et al.* (2011)

and the subject categories presented during Twitter's users sign up process, the following verticals were created: *general, politics, entertainment, technology, breaking, movies, science, sports, music*. Each of these sources was then assigned to one of multiple vertical according to the mapping shown in Table 3.1.



(a) TREC 2013.



(b) TREC 2014.

Figure 3.3: Relevant documents in the 10% sample by topic and vertical.

3.2.3 NewsSources relevance judgments

Relevance judgments are needed for the evaluation of resource selection algorithms, especially in terms of recall. They were obtained for the top 10 documents retrieved in response to the TREC 2013 and 2014 topics using a query-likelihood retrieval model, language modeling with Dirichlet smoothing ($\mu = 2500$), over the CSI collection.

The documents were assessed by workers in the CrowdFlower crowdsourcing platform until the inter-annotator agreement reached 0.7 for each work item. Relevance judgments were obtained on a three-point scale: not relevant, relevant, and highly relevant. Workers had clear instructions and followed a number of simple rules:

- Examine the tweet text and look for person names, and other types of entities. Take special attention to hashtags, which can be used to refer to a specific event.
- Follow the links in the tweet, as they can be crucial in deciding what relevance level to attribute to the tweet.
- Search the Web using the provided Google search button to research the search topic. Make additional searches on other search engines, Wikipedia, news articles, and other authoritative websites, if needed.

Figure 3.3, shows the distribution of relevant documents grouped by TREC topic and vertical. The *HuffingtonPost* was the top source with 45 relevant documents across various TREC queries, followed by *Reuters* and *USA Today* with 26 and 20 relevant documents, respectively. Note that the *HuffingtonPost* was also the source with the highest number of documents by a significant margin (see Figure 3.2b).

The relevance judgments collected via crowdsourcing also helped to estimate if the expansion corpus had sufficient coverage for the TREC topics. The final set of news-related queries that is used in the experiments contains 27 topics from TREC 2013 dataset and 27 topics from the TREC 2014 dataset.

3.2.4 Methods

No-PRF is a QL retrieval model with Dirichlet smoothing ($\mu = 2500$).

CLRM Condensed List Relevance Models, recently proposed by Diaz (2015) essentially re-ranks the list of results retrieved by the initial query using the expanded query generated with the same list using relevance models.

PRF.wiki is a pseudo-relevance feedback baseline that uses an external index of the English Wikipedia article pages XML. It was used for expansion in blog search by Arguello *et al.* (2008) with significant improvements in effectiveness. The *snapshot* (*dump*)[1] processed, is dated from just before the period covered by the microblog evaluation dataset, to avoid using future evidence. The snapshot of Wikipedia articles was first preprocessed using *wikiextractor*[2] to obtain clear indexable text. For PRF.wiki the 10 articles were used for feedback.

PRF is a standard pseudo-relevance feedback method that uses the whole search index for feedback. The RM3 pseudo-relevance feedback algorithm (Lavrenko and Croft, 2001) is used for all the methods based on PRF because it was shown to be very effective in previous microblog retrieval research and it has similar information requirements and computational characteristics to other PRF algorithms. In all the PRF based methods, the top 50 documents retrieved for each query q using the *query-likelihood* retrieval model were used for feedback, language modeling with Dirichlet smoothing ($\mu = 2500$).

PRF.news Our retrieval baseline for query expansion using the *NewsSources* dataset is a sample-based method, which corresponds to pseudo-relevance feedback over an index of the complete *NewsSources* collection.

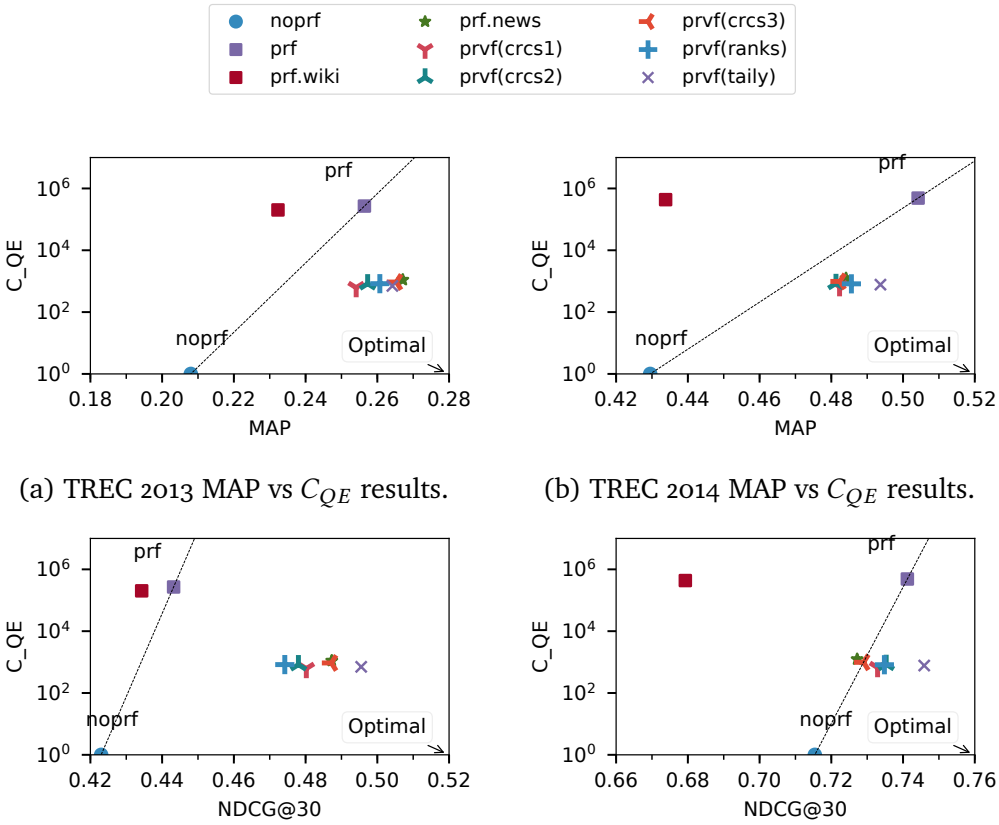
PRVF. Three variants of PRVF corresponding to the different resource selection algorithms employed: CRCS, Rank-S, or Taily.

PRVF (crcs) does not dynamically adjust the number of verticals selected, therefore it is evaluated at three fixed limits for the number of verticals selected: $|\mathscr{C}_{q)}| = \{1, 2, 3\}$. For instance, PRVF (crcs1) corresponds to expanding the queries using only the top vertical selected by CRCS.

[1] enwiki-20130102-pages-articles.xml
[2] https://github.com/attardi/wikiextractor

PRVF (taily) dynamically adjusts the number of selected verticals $|\mathscr{C}_q|$ for each query. The Taily resource selection algorithm was parameterized with the values ($n = 400$, $v = 50$) recommended in (Aly *et al.*, 2013).

PRVF (ranks) dynamically adjusts the number of selected verticals $|\mathscr{C}_q|$ for each query. We set the threshold $minRanks = 1e^{-6}$.



(a) TREC 2013 MAP vs C_{QE} results.

(b) TREC 2014 MAP vs C_{QE} results.

(c) TREC 2013 NDCG@30 vs C_{QE} results.

(d) TREC 2014 NDCG@30 vs C_{QE} results.

Figure 3.4: MAP and NDCG@30 vs C_{QE} in TREC 2013 and TREC 2014.

3.3 Results and Discussion

The evaluation is organized as follows: firstly, an analysis of the efficiency (Subsection 3.3.1) and effectiveness (Subsection 3.3.3) of the PRVF and PRF.news

methods. Secondly, an analysis of the recall of different resource selection algorithms that led to the choice of partitioning by news verticals (Subsection 3.3.5).

Finally, the standard implementation of PRF, based on re-retrieval, was compared with Condensed List Relevance Models (CLRM), an implementation based on re-ranking, recently proposed by Diaz (2015) (Subsection 3.3.4).

3.3.1 Efficiency Analysis of PRF methods

In this study, the focus is on the computational cost of the initial retrieval, necessary for PRF-based query expansion. In this section, the computational cost of query expansion, C_{QE}, is measured as the number of posting lists accessed in the query expansion process. Subsection 3.1.4 presents the theoretical expected costs of query expansion for each method. Figure 3.4 shows the trade-offs between the cost, C_{QE}, and the corresponding retrieval metrics results on the TREC Microblog datasets. C_{QE} is represented in the y-axis in log-scale to get an overview of how the proposed approach compares in terms of efficiency to No-PRF and the standard PRF baselines. In the x-axis, there is either the MAP or the NDCG@30 retrieval metric. Since the objective is to lower C_{QE} to be more effective, the desired method would fall below the dashed line that goes from No-PRF to PRF, and towards the bottom right corner.

Response times were measured using C_{Lat}, which gives us the largest amount of work done by any vertical in a parallel search scenario. We find that query expansion response times are halved in both datasets compared to PRF.news.

Pseudo-Relevant Vertical Feedback. All PRVF methods had a lower computational cost C_{QE} than PRF.news and are clustered around the same area in the graphs. Even though the cost of the PRVF methods was considerably lower, the results were similar to PRF.news in terms of MAP. Therefore, PRF.news seems to be a good predictor for the expected retrieval effectiveness with the PRVF architecture (see Figure 3.4b).

The NDCG@30 results obtained with PRVF methods are similar or better than PRF (see Figure 3.4d). The intuition is that more computational cost should translate into a better ranking. However, some PRVF methods provided

a better top rank with lower computational cost. PRVF (taily) outperformed the other methods in NDCG@30 in the TREC 2014 queries. However, on the TREC 2013 queries PRVF (crcs3) outperformed the PRVF (taily) method slightly (see Figure 3.4c).

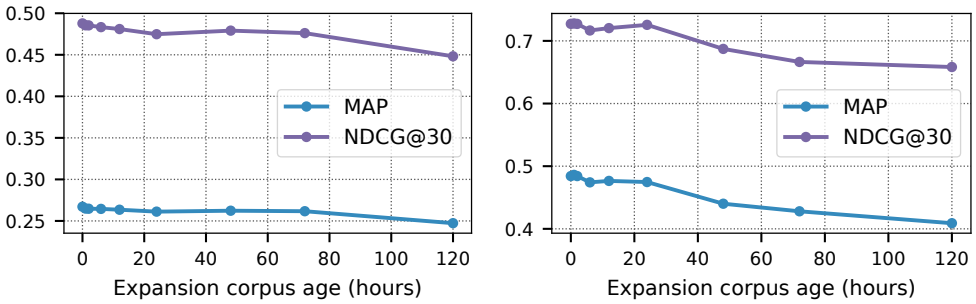
All PRVF-based were three orders of magnitude less computationally expensive than the PRF baseline. The PRVF-based methods were the most efficient since for each query they only search the most promising verticals. Query expansion response times were cut in half on both datasets compared to the PRF.news baseline as measured by C_{Lat}, which takes into account the parallelism afforded by a distributed architecture.

CLRM Condensed List Relevance Models is a new query expansion approach, based on relevance models, and recently proposed by Diaz (2015). Essentially, it re-ranks the initial list of results retrieved by the initial query using the expanded query generated with the same initial list of results. The re-retrieval step is avoided, which is a significant advantage of this method. Its retrieval cost amounts to the cost of the initial retrieval, which is expected to be lower than the cost of the re-retrieval step.

3.3.2 Quality of Expansion Corpus

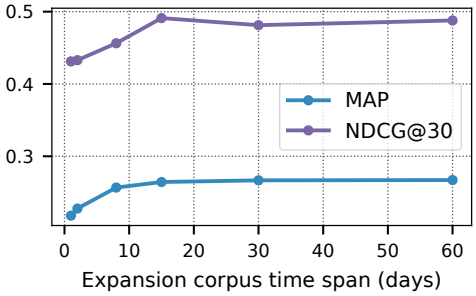
In this section, we analyze potential biases in the vertical expansion corpus and the importance of the expansion corpus age and time span. Since PRVF uses documents from news sources for query expansion, we might improve the chances of retrieving tweets from these sources. To make sure that bias is not improving the results unfairly we counted the number of documents marked as relevant in the main index (*TREC 2013 and 2014*) that are in the expansion corpus (*NewsSources*). The overlap was of only nine relevant documents in TREC 2013 and thirteen relevant documents for TREC 2014. Thus, we did not find any evidence that the choice of news sources affords any unfair advantage.

A key aspect of the PRVF architecture is its ability to cope with multiple information streams that are constantly feeding the query expansion corpus. In Figure 3.5a and Figure 3.5b we observe how the *expansion corpus age* (i.e.,

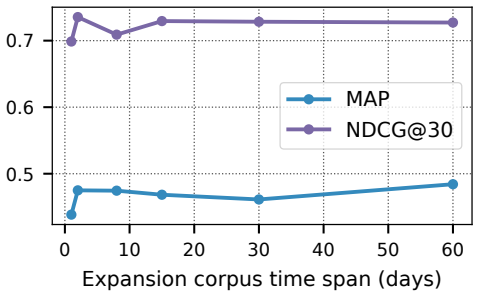


(a) TREC 2013.

(b) TREC 2014.



(c) TREC 2013.



(d) TREC 2014.

Figure 3.5: Analysis of expansion corpus age and time span.

the difference between queries timestamp and the most recent document timestamp), is clearly linked to the decay in retrieval precision. The time span of the expansion corpus is also examined in Figure 3.5c and 3.5d – here we can observe that it might be sufficient to keep only the last 15 days for query expansion. This fact confirms the initial assumption that in Microblog search it is critical to use an up-to-date expansion corpus. Besides, we found that it is not necessarily to keep an expansion corpus with a long time span.

3.3.3 Retrieval Effectiveness of PRF Methods

More detailed results of our evaluation over the two sets of queries are presented for TREC 2013 (Table 3.3) and TREC 2014 (Table 3.4). The tables contain the average over all queries for the retrieval metrics MAP and NDCG@30. The average computational cost of each approach over all queries is presented in the C_{QE} column. After the cost of query expansion C_{QE}, in parentheses, we show

Table 3.3: TREC 2013 dataset results.

	C_{Lat}	C_{QE}	MAP	NDCG
w/o External Expansion Corpus				
No-PRF	0	0	0.2080	0.4230
PRF	269175	269175	0.2564	0.4432
w/ External Expansion Corpus				
PRF.wiki	201892	201892	0.2323	0.4343
PRF.news	1110	1110	**0.2671**‡	0.4873‡
w/ External Vertical Expansion Corpus				
PRVF (crcs1)	616 -43.2%	616 -43.2%	0.2541‡	0.4802‡
PRVF (crcs2)	653 -41.2%	848 -31.6%	0.2573‡	0.4780‡
PRVF (crcs3)	655 -41.0%	982 -20.7%	0.2653‡	0.4872‡
PRVF (ranks)	673 -39.4%	821 -33.7%	0.2607‡	0.4742†
PRVF (taily)	**509** -54.1%	**773** -37.6%	0.2642‡	**0.4955**‡

Symbols † and ‡ stand for a statistically non-inferior result to PRF with $p < 0.05$ and $p < 0.01$ respectively, according to a non-inferiority test (Walker and Nowacki, 2011).

cost reduction in relation to PRF.news.

Firstly, the results for the Wikipedia baseline **PRF.wiki**. Using a recent Wikipedia corpus for feedback provided an improvement of MAP on the TREC 2013 queries only. In the TREC 2014 queries, NDCG@30 was 5.3% lower than the No-PRF baseline. Also, even though the Wikipedia corpus is smaller than the target retrieval corpus, the average computational costs were still very high, at 202k and 434k accessed postings for TREC 2013 and TREC 2014, respectively. Therefore, using Wikipedia for query expansion in microblog search was found to harm retrieval effectiveness metrics maybe because the expansion collection is not kept strictly up-to-date.

The PRF baseline has significantly stronger results in the retrieval effectiveness metrics than No-PRF and PRF.wiki. In the TREC 2013 queries, the PRF baseline improved on MAP over No-PRF by 18.9%, a statistically significant improvement. NDCG@30 was also improved by 4.6%. In the TREC 2014 queries

Table 3.4: TREC 2014 dataset results.

	C_{Lat}	C_{QE}	MAP	NDCG
w/o External Expansion Corpus				
No-PRF	0	0	0.4295	0.7154
PRF	487547	487547	**0.5042**	0.7412
w/ External Expansion Corpus				
PRF.wiki	434233	434233	0.4338	0.6793
PRF.news	1239	1239	0.4841	$0.7272^{\dagger}$
w/ External Vertical Expansion Corpus				
PRVF (crcs1)	661 -46.7%	661 -46.7%	0.4823	$0.7329^{\ddagger}$
PRVF (crcs2)	734 -40.1%	848 -31.6%	0.4813	$0.7353^{\ddagger}$
PRVF (crcs3)	734 -40.1%	982 -20.7%	0.4824	$0.7290^{\dagger}$
PRVF (ranks)	734 -40.1%	821 -33.7%	0.4856	$0.7348^{\ddagger}$
PRVF (taily)	**575** -53.6%	**773** -37.6%	$0.4927^{\ddagger}$	$\mathbf{0.7470}^{\ddagger}$

Symbols † and ‡ stand for a statistically non-inferior result to PRF with $p < 0.05$ and $p < 0.01$ respectively, according to a non-inferiority test (Walker and Nowacki, 2011).

PRF improved on MAP over No-PRF by 14.8%, a statistically significant improvement. NDGC@30 improved by 3.5% over No-PRF. However, the average cost of query expansion using standard PRF was very high for TREC 2013 and TREC 2014 with 269k and 488k accessed postings, respectively.

Compared to the best run submitted by participants of TREC 2013, PRF.trec2013, PRVF was able to match MAP and improve on NDCG@30. Our methods outperformed PRF.trec2014 on the TREC 2014 topics in both metrics.

The PRVF (taily) method was the most balanced in the TREC 2013 queries with a computational cost of only $C_{QE} = 703$, which corresponds to a cost reduction of 36.7% over PRF.news, or around 1.6× faster. It had one of the highest MAP results (3.0% higher than PRF) and improved 21.3% over No-PRF (statistically significant). It also had the second-best NDCG@30 result, improving 1.7% over PRF.news and 10.6% over standard PRF.

PRVF (taily) provided the best balance for the TREC 2014 queries as well. It obtained a 12.8% improvement in MAP over No-PRF (statistically significant) and

a 4.2% improvement in NDCG@30 with a computational cost of only $C_{QE} = 773$. PRVF (taily) was 1.6× faster than searching the whole news index PRF.news a cost reduction of 37.6%. The top MAP on the TREC 2013 queries was obtained with PRVF (crcs3). Because a fixed number of verticals are searched for each query (3), which is higher than PRVF (taily)'s average, the cost reduction was smaller (15.5% over PRF.news).

3.3.4 Re-Ranking PRF and Short Text Documents

Table 3.7 presents retrieval effectiveness metrics for CLRM and other PRF implementations based on re-retrieval. The table presents a set-based retrieval metric *set_recall*, which corresponds to the percentage of relevant documents retrieved in the top 1000 results. Not to be confused with P@1000. CLRM outperforms No-PRF in both MAP and NDCG@30.

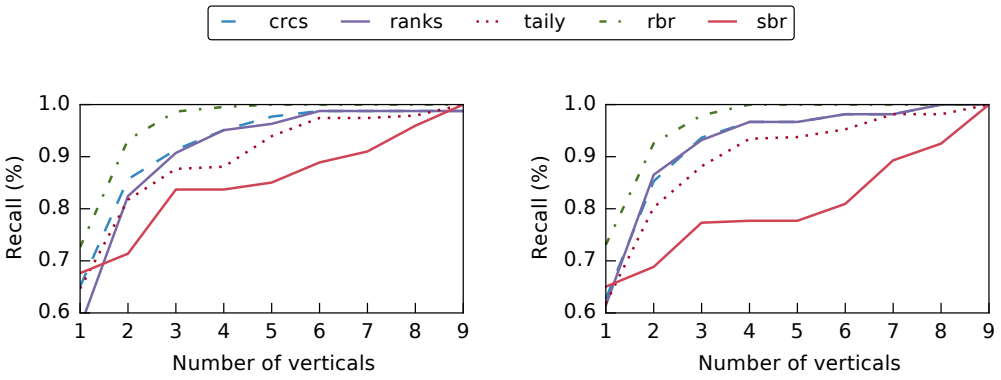
However, since CLRM just re-ranks the documents retrieved by the initial query, its *set_recall* is the same as the query-likelihood baseline No-PRF. Due to this, CLRM was not able to achieve the same retrieval effectiveness of the implementations based on re-retrieval. Note that the generated expanded query is the same for both CLRM and PRF. However, since PRF does a re-retrieval it was more effective.

In short text document indexes, some relevant documents that are ranked at the top by a re-retrieval implementation might be missing from the initial retrieval using the original query terms only. In addition, some relevant documents might contain only a few of the original query terms, a problem that is exacerbated by the short size of the documents in a microblog corpus. Therefore, in short text datasets, an implementation of pseudo-relevance feedback based on re-retrieval might be preferred to achieve similar retrieval effectiveness.

The PRF.wiki method, based on re-retrieval, was able to retrieve more relevant documents than CLRM with a higher set_recall in both datasets. However, the higher recall did not translate into better search results since the generated query expansions using Wikipedia feedback were less effective for ranking.

Our PRVF (taily) approach generates query expansions using a more efficient

federated query expansion architecture over an external news corpus. The quality of the generated query expansions using this method can be attested from its set_recall and better effectiveness metrics compared with PRF.



(a) TREC 2013.

(b) TREC 2014.

Figure 3.6: Recall on the query expansion corpus using verticals.

3.3.5 Selection Methods Recall Analysis

In this section, we analyze the impact of different state-of-the-art resource selection algorithms in terms of recall. In order to generate better expansion terms in PRF, a few quality documents need to be retrieved in the initial retrieval step. To reduce the cost of feedback, the proposed approach leverages on resource selection algorithms to select the most promising verticals, on a per-query basis. A sufficient amount of feedback documents needs to be retrieved from the selected verticals to build a good relevance model. The sample-document methods, CRCS and Rank-S, use a CSI with a sample of ~12% the size of the *NewsSources* collection. The effectiveness of resource selection algorithms can be hurt by using small samples. Vocabulary-based selection algorithms seem to be more affected than sample-based methods, and therefore Taily's statistics database was built using the *NewsSources* collection to get a similar performance to sample-document methods. All resource selection algorithms used the *query-likelihood* retrieval model, language modeling with

Dirichlet smoothing ($\mu = 2500$), and the Indri English stop words list for indexing. All methods were able to reach a recall higher than 80% using just the first two verticals selected.

Therefore, the dynamic cutoff threshold for Rank-S and Taily were tuned accordingly. Taily dynamically limited the selected verticals to 2.19 verticals on average for both sets of queries. Rank-S selected 2.22 and 1.81 verticals on average for the TREC 2013 and TREC 2014 queries, respectively. Some variance in the number of times each vertical was selected for expansion was expected when using different resource selection algorithms. The most striking difference is that for TREC 2013 the *entertainment* vertical was selected up to 3× more when using Taily (see Table 3.5).

Table 3.5: Selected verticals TREC 2013.

Vertical	Taily	Rank-S	CRCS2
general	26	25	25
politics	11	7	5
technology	6	4	3
entertainment	3	10	7
movies	4	3	3
music	3	0	1
science	3	6	5
sports	1	3	2
breaking	2	2	3
$avg(\|\mathcal{C}_q\|)$	2.19	2.22	2

Figure 3.6 shows the recall profile for the different resource selection algorithms considered. This graph shows how well each resource selection algorithm selected the top verticals, the ones which could lead to the retrieval of more relevant documents. A Relevance-Based Ranking upper-bound **(RBR)** and a Size-Based Ranking lower-bound **(SBR)** were included in these graphs for comparison. We defined the **RBR** method as an upper-bound: for a given query, it selects the verticals with the highest number of relevant documents, according to the relevance judgments obtained for *NewsSources*. The lower-bound method

Table 3.6: Selected verticals TREC 2014.

Vertical	Taily	Rank-S	CRCS2
general	25	25	26
politics	10	7	8
technology	3	2	4
entertainment	6	4	4
movies	4	3	3
music	0	0	2
science	3	3	4
sports	1	0	0
breaking	7	5	3
$avg(\|\mathscr{C}_q\|)$	2.19	1.81	2

SBR corresponds to a size-only selection heuristic: for a given query, verticals are selected in order by number of total documents, from largest to smallest (see Figure 3.2a).

CRCS and Rank-S had better recall profiles. In both TREC 2013 and TREC 2014 query sets, they provided more than 80% recall with just the top two verticals (see Figure 3.6). Taily was not as successful as the sample-document methods in terms of recall, especially after the first three verticals. Overall, CRCS had the best recall at one. It had a slightly better recall profile, selecting verticals with a higher number of relevant documents earlier.

Table 3.7: Retrieval results using CLRM on microblog datasets.

	TREC 2013			TREC 2014		
	MAP	NDCG	Recall	MAP	NDCG	Recall
No-PRF	0.2080	0.4230	0.5188	0.4295	0.7154	0.6994
PRF	0.2564	0.4432	0.5764	**0.5042**	0.7412	**0.7860**
CLRM	0.2276	0.4423	0.5188	0.4718	0.7416	0.6994
PRF.wiki	0.2323	0.4343	0.5689	0.4338	0.6793	0.7443
PRVF (taily)	**0.2642**	**0.4955**	**0.5921**	0.4927	**0.7470**	0.7818

3.4 Summary

This chapter presented an efficient method for pseudo-relevance feedback in microblog search that uses news published on Twitter as query expansion corpus. The architecture proposed organizes a large collection of documents, published by a set of news sources into news verticals. It requires a curated set of sources and its assignments to verticals, which can be difficult to obtain for some domains. However, it is easily extensible, by increasing the number of sources in each vertical (to provide stronger coverage), and more verticals can be added (to provide broader coverage).

The federated query expansion approach delivered a retrieval effectiveness similar to standard PRF at a fraction of the computational cost. First, the Twitter news are partitioned into vertical index shards and leveraged on both sample-based and vocabulary-based resource selection algorithms to select and search only the most promising verticals. We found that a federated query expansion approach provided similar retrieval effectiveness to query expansion using the whole news corpus at a much lower computational cost. Experiments using Wikipedia as the feedback corpus showed that computational costs are not reduced significantly and was not as effective. The evaluation of the proposed architecture led to the following concluding points:

- ***Federated QE.*** The proposed PRVF method is an efficient federated query expansion architecture for microblog search, where the expansion corpus is live and new documents are arriving from different news sources in streaming fashion. Partitioning news sources into verticals and using pseudo-relevant vertical feedback (PRVF) methods delivered the most **efficient QE**. Furthermore, this approach outperformed the retrieval effectiveness of using the non-partitioned news index (PRF.news) and also the whole search index (PRF).

- ***Cost-effective PRF.*** The best balance between efficiency and effectiveness was obtained using PRVF (taily), which was more robust than other approaches while using on average fewer verticals. PRVF (taily) achieved

the highest results in effectiveness metrics for both the TREC 2013 and TREC 2014 query sets, except for MAP on TREC 2013, which was slightly higher with PRVF (crcs3). These results indicate that resource selection algorithms that can limit the number of verticals searched dynamically are more suitable for this task.

In the next chapter we present a framework that mines the behavior of the crowds for additional temporal signals. Our novel time-aware ranking method integrates lexical, domain, and temporal evidences from Wikipedia (through page views and page edit history), news articles, and Twitter feedback.

MINING TEMPORAL RELEVANCE FROM MULTIPLE SOURCES FOR TIME-SENSITIVE QUERIES

4

"There will always be plenty of things to compute in the detailed affairs of millions of people doing complicated things."

— Vannevar Bush

In social media, and more so in microblog social networks, the generation of new content by the users reflects the collective attention of the crowd. People turn to social media to post about breaking news and trending topics generating a higher activity related to a topic on the Web. The propagation of new information through multiple mediums and social media prompts this phenomenon.

User-generated content platforms offer new rich data sources for analyzing temporal patterns of information creation and information seeking. Crowd-sourced content generation platforms, such as Wikipedia, are a great example of the temporal dynamics on the Web. In one hand, searchers look for more information on the Web generating an unusual demand for specific articles on Wikipedia (Ciglan and Nørvåg, 2010). On the other hand, early on, users edit the articles on Wikipedia referring to new events (Ferron and Massa, 2012).

Each one of these phenomena often occur within a small window of time from the event. People can discuss an event *during* its occurrence, (e.g., televised live sports broadcast), *after* learning new information, (e.g., election results), or even *before*, as we will see, in anticipation of a scheduled event (e.g., the Olympic Games and the Oscars).

Major events are reported on by the traditional news media outlets, which now also publish their news stories online and in real-time. People looking to learn more about an event might use a Web search engine to find information, generating an abnormal burst of interest for a certain query.

In this chapter, we propose to estimate the relevant time periods for a query by leveraging on temporal patterns of user behavior gathered from external information sources. We propose a novel time-aware ranking model that leverages on multiple sources of temporal signals to provide a more robust estimation of the relevant periods for a query. It is hypothesized that, the chances of finding relevant documents are higher on the *peak times* of interest towards an event.

The analysis of multiple sources of temporal signals is a powerful tool that can allow us to stitch together different pieces of evidence to improve the estimation of temporal relevance. Previous work has typically gathered temporal signals from a single source, usually the corpus itself via feedback. However, these strategies assume that events have a homogeneous impact across all information sources and that the temporal signal available from a single source will be clear and unambiguous.

4.1 Barbara Made the News

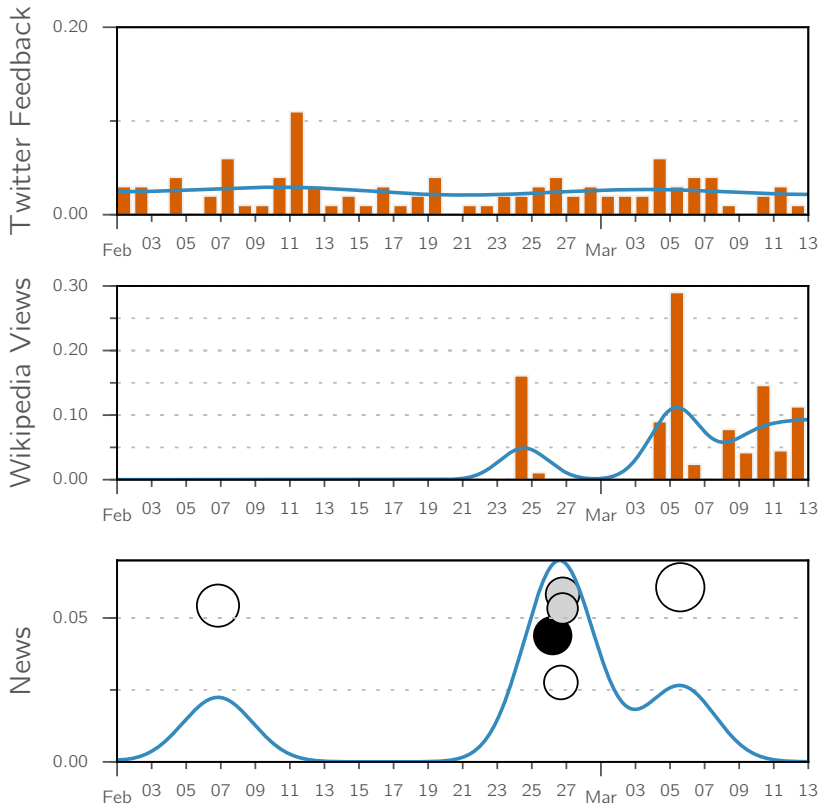
On February 2013 it was revealed that Barbara Walters, a well-known figure in U.S. television, got chicken pox. This event sparked multiple processes on the Web such as the propagation of news articles about the event and increased interest in the article "Barbara Walters" on Wikipedia. These temporal signals can be mined in real-time, as the event unfolds and can also be mined retrospectively for some Web sources. In Figure 4.1 we dissect how measurable activity on the Web, generated in reaction to this newsworthy event, displayed temporal patterns related to the Web crowd behavior. By looking at the distribution of initially retrieved documents from Twitter, when searching for "Barbara Walters, chicken pox," it was not possible to intuit the time periods that might contain relevant documents. Moreover, there seemed to be no correlation with the known distribution of relevant tweets. Using the collection itself as a single source evidence (i.e., Twitter Feedback), did not allow the extraction of useful temporal signals. In this case, it is necessary to look into other sources that can help disambiguate the relevant time periods for this query.

The *Wikipedia views* temporal signal graph depicts the daily volume of page views and its evolution for the article "Barbara Walters" on Wikipedia. It is easy to see that the largest spike of page views occurred on March 4 and then the volume of page views declines in the following days as interest in the topic subsides. The temporal profile of Wikipedia views loosely matches the days that contain the highest volume of relevant documents in the relevance judgments.

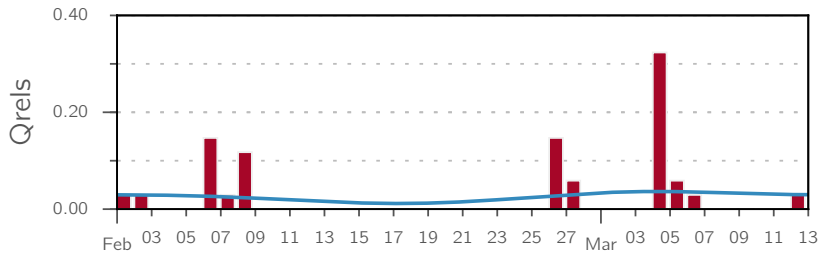
Using *News* sources there are three main time periods that could be extracted, which seem to model the temporal relevance of the topic better. A first peak corresponds to the publication time of a news article reporting on an outbreak of *chicken pox* that mentions *Barbara Walters*. The second peak corresponds to the publication of multiple news articles reporting that Barbara Walters was recovering at the time. Finally, the third peak corresponds to the publication time of a news article reporting the news of her return to U.S. television talk show "The View."

The estimated relevant time periods as predicted by each source, can be compared to the actual temporal distribution of relevant documents in the ground-truth Figure 4.1b. One key insight from this comparison is that two of the sources contain useful temporal information and provide different relevant time periods. This analysis hints at the need for mining multiple sources of evidence to estimate temporal relevance on Twitter. This is more robust and diverse than using a single source of evidence.

The rationale supporting this research hypothesis calls for a method that overcomes this problem by disambiguating the relevant time periods using multiple sources of evidence: internal and external, to improve retrieval accuracy. A novel time-aware ranking method that combines evidence from multiple temporal sources, Ranking with Multiple Temporal Sources (RMTS), is proposed.



(a) Temporal evidence from multiple Web sources.



(b) Temporal distribution of relevant tweets (qrels).

Figure 4.1: The query "Barbara Walters, chicken pox" according to sources. The histograms indicate the activity in each source per day. In the News source, each circle corresponds to a matching news article, its size is proportional to the Jaccard index between the query and the news headline. Its color indicates the news site (Reuters – gray, Associated Press – black and USA Today – white).

4.2 Temporal Signals

Events have an impact not only on social media but also on Wikipedia page views and edits, and the news. This chapter departs from previous approaches that use a single source and investigates the use of multiple sources of information to provide several independent temporal signals and broader coverage. It relies on the intuition that discussions about an event (and therefore relevant documents) are more likely to occur around the same time across different information sources. Leveraging on multiple external data sources is advantageous since they can offer more information to estimate the temporal query intent. The first step towards a solution requires the design of a unified representation of temporal signals that given query q, mines and represents temporal evidence from multiple sources of temporal signals.

Temporal queries are the ones where relevance is time-dependent. These queries often exhibit temporal search patterns that can be measured in query logs. However, some temporal queries might exhibit no temporal search patterns but still have underlying time-dependence. In the same way, temporal queries might not exhibit temporal patterns in feedback documents used for temporal feedback (Efron *et al.*, 2014). However, temporal patterns might be observed in external data sources that are linked in some way to the temporal queries. Therefore, temporal patterns in external data sources can help refine the estimation of temporal relevance for a query and, in some cases, be the only sources that exhibit temporal patterns. This approach brings a series of novel contributions:

1. the mining of temporal signals from different external sources;
2. a unifying representation of temporal relevance;
3. a time-aware ranking model.

4.2.1 A Unified Representation of Temporal Signals

We first define $S = \{s_1 s_2 \cdots s_{|S|}\}$ as the set of external sources of temporal information that are expected to reflect temporal patterns of user behavior

towards an event. The goal is to be able to estimate the relevance of instant t for a given query q. We want to estimate a function

$$f_{s_x}(q,t) \in [0,1] \tag{4.1}$$

that for each source s_x, computes the probability of relevance of instant t for query q. This t will be formalized as the timestamp of a document t_d.

To estimate f_{s_x}, the data needs to be obtained from source s_x. In this chapter, a temporal signal corresponds to data mined from an information source that provides temporal patterns of the collective activity of users. Once obtained, an information extraction process generates two paired sequences of information: a set of instants T, and λ a set of instant weights, formally

$$T^{s_x} = \{t_1^{s_x} t_2^{s_x} \cdots t_n^{s_x}\} \quad \lambda^{s_x} = \{\lambda_1^{s_x} \lambda_2^{s_x} \cdots \lambda_n^{s_x}\} \tag{4.2}$$

The pair $\langle t_i^{s_x}, \lambda_i^{s_x} \rangle$ fully characterizes a temporal signal – its timestamp t and its importance λ to the query at hand. This notation is used throughout the chapter for any external data source.

For each document d we would like to find $P(r \mid t_d, q, s_x)$, the probability of relevance of its timestamp t_d according to source s_x and the query q. The estimation of the joint distribution $f_{s_x}(q,t)$ over the time span of the corpus is key. The probability density function $f_{s_x}(q,t)$ is estimated using the kernel density estimation method which is advantageous due to the natural smoothness of the final estimated function:

$$\hat{f_{s_x}}(t) = \frac{1}{nh} \sum_{i=0}^{n} \lambda_i^{s_x} K\left(\frac{t - t_i^{s_x}}{h}\right) \tag{4.3}$$

where $t_i^{s_x}$ are the timestamps mined from a source s_x, the kernel function $K(z)$ corresponds to the Gaussian kernel $\mathcal{N}(z,0)$, and the optimal bandwidth can be estimated by a data-driven method such as Silverman's rule-of-thumb $h^* \approxeq 1.06\,\hat{\sigma}\,n^{-1/5}$. Finally, $\lambda_i^{s_x}$, is a non-negative weight on timestamp $t_i^{s_x}$, to weight it by importance.

4.2.2 Temporal Signals from Multiple Sources

Microblog queries can have *peak times* following events, related to breaking news, celebrities, other entities and events, periodic queries, e.g., TV shows, and ongoing events. The assumption underlying this approach is that topics that burst on microblogs are correlated with a higher volume of page views and edits for related Wikipedia articles.

To extract temporal signals that match the search query, one needs to design source-specific methods to filter candidate temporal signals and compute their importance to the search query.

4.2.2.1 Twitter

Recent works have used temporal feedback for time-aware ranking in *tweet* search (Dakka *et al.*, 2012; Efron *et al.*, 2014). Temporal feedback consists in using the temporal distribution of the documents retrieved by a standard retrieval model to estimate temporal relevance. The temporal signal provided by temporal feedback, using retrieval over the collection of Twitter posts, is defined as source s_f.

These methods are rooted in the assumption that search query results will originate two distinct distributions, a lexical and temporal distribution, that must be integrated into a single rank. The set of temporal signals are extracted from tweets retrieved with query q using a standard retrieval model – the temporal signals $T^{s_f} = \{t_1^{s_f} t_2^{s_f} \cdots t_n^{s_f}\}$, a collection of the timestamps of the tweets retrieved. The weight given to each timestamp $\lambda_i^{s_f}$ is calculated according to the textual similarity score

$$\lambda_i^{s_f} = \frac{P(q \mid d_i)}{\sum_{j=1}^{N} P(q \mid d_j)} \tag{4.4}$$

4.2.2.2 News

News headlines produced by multiple news sources can be indexed incrementally and in real-time. In this chapter, news headlines were obtained using a Web

search engine API over a set of high signal-to-noise ratio news sources: the Associated Press (AP), BBC's UK and World (BBC), Reuters, and USA Today. Therefore, to find the publication date of news articles an automatic rule-based extraction method was used to extract explicit non-relative temporal expressions (Schilder and Habel, 2003). Extracted times were converted to UTC.

For a given query q, the Web search engine is queried and a set of news headlines matching the search query is returned. The set of candidate headlines

$$L = \{l_1 l_2 \cdots l_k\} \tag{4.5}$$

is retrieved without document scores since this Web search engine API omits them. The temporal signals of news source s_l are represented by the set of headline timestamps T^{sl}.

The timestamp of each headline $t_i^{s_l}$ should be weighted according to its relevance or match to the query. For this purpose, the Jaccard similarity coefficient was used to measure the similarity between a headline and the query. This choice seems appropriate since for both headlines and queries the frequency of any word is often $\text{tf}(w_j \in q) = 1$. On the other hand, it can also avoid over-weighting duplicate words. The Jaccard similarity coefficient between two sets of words A and B is given by

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.6}$$

Thus, the weight λ_i^{sl} is computed as the Jaccard similarity $J((, q), l_i)$.

4.2.2.3 Wikipedia

To model the users' behavior, We opted to use the Wikipedia article search functionality to find related one related Wikipedia article for each query. We hypothesize that users enter similar queries when searching on Twitter and on the Web and this approach models this user behavior. Following this rationale, we posit that: some unrelated Wikipedia article pages that match the query

should still exhibit temporal patterns, especially page views, since users are likely to visit these when searching the Web for more information.

For each query q, submitted to the Wikipedia Search API, the first 10 article page titles were retrieved and re-ranked using the Jaccard similarity coefficient. The most similar article page title was selected. Several works use entity linking methods or APIs, such as the TAGME (Ferragina and Scaiella, 2010), to link text to a Wikipedia article page. In this framework, temporal signals are extracted from a single Wikipedia article page, thus entity linking methods could have been used.

Wikipedia page views. Wikipedia article page views statistics can be extracted from Wikipedia's Web server access logs that the Wikimedia project makes publicly available packaged into hourly rolling tarballs. The public API[1] was used because it provided histograms of page views in daily bins, which we will refer to as source s_v.

Temporal signals were extracted from page views statistics for the selected Wikipedia article fetched from 1 December 2012 up until query time, hence up to 4 months of data was fetched. To reduce noise, each article page views are normalized using the mean daily page views count for that article. The normalized set of daily page views counts was used as weight $\lambda^{s_v} = \{\lambda_1^{s_v} \lambda_2^{s_v} \cdots \lambda_n^{s_v}\}$ for the corresponding days $T^{s_v} = \{t_1^{s_v} t_2^{s_v} \cdots t_n^{s_v}\}$, where the timestamp used was set to midnight UTC.

Wikipedia page edits. The Special:Export[2] page allows exporting the latest revisions of Wikipedia articles to a standardized XML format. For this temporal feature, page edits from source s_e, the latest revisions for the Wikipedia article were downloaded. The XML response was processed to create a *diff* of the changes between each consecutive pair of article revisions. Automated article changes made by known Wikipedia Bots[3] were filtered out.

For each temporal signal, a revision timestamp t_{e_i}, was weighted by term frequency, the sum of the frequency of query terms in the added text of e_i. This

[1] http://stats.grok.se/
[2] https://en.wikipedia.org/wiki/Special:Export
[3] https://en.wikipedia.org/wiki/Wikipedia:Bots

approach aims to boost changes to Wikipedia articles, which have a stronger evidence of matching the query. Terms contained in the title of the selected Wikipedia article were used as stop words. For instance, if the article selected is "Barbara Walters" only changes that match something other than either "Barbara" or "Walters" were counted towards the frequency-based weight. The temporal signals extracted from page edits is given by $T^{s_e} = \{t_1^{s_e} t_2^{s_e} \cdots t_n^{s_e}\}$. The weight of each timestamp is the sum of the frequency of terms shared by the query and added in the revision as follows

$$\lambda_i^{s_e} = \sum_{j=1}^{|q|} \text{tf}(q_j \in e_i) \tag{4.7}$$

4.3 Ranking Framework

Our model follows a learning to rank (LTR) framework that integrates text features, domain-specific features (e.g., number of hashtags) and temporal features extracted from crowd behaviors. In this section, we will first discuss the learning to rank model, followed by the description of the set of non-temporal features, and then address the computation of the model parameters.

4.3.1 Ranking with Multiple Temporal Signals

Typical LTR models consider a diverse set of features from the corpus. In this study, the features were integrated using a linear model, where the retrieval score of document d for a given query q is given by

$$LTR(q,d) = \sum_i \alpha_i f_{l_i}(q,d) + \sum_j \beta_j f_{c_j}(d) \tag{4.8}$$

The set of lexical features, f_{l_i} covers several text statistics and retrieval scores. Learning to rank literature has also extensively shown that non-lexical features f_{c_j}, such as number of tweets and number followers, were essential to capture the relevance of a document.

In the previous section, we saw how temporal evidence could be mined from data generated by the crowds. In Table 4.2 the set of temporal feature signals is summarized. We proposed a unified view of how temporal signals can be represented. Following this reasoning, we are now ready to plug-in multiple temporal evidence into the initial retrieval model, by extending it to support the time dimension:

$$RTMS(q, t_d, d) = \underbrace{\sum_i \alpha_i f_{l_i}(q, d)}_{\text{query-dependent}} + \underbrace{\sum_j \beta_j f_{c_j}(d)}_{\text{query-independent}} + \underbrace{\sum_x \gamma_x f_{s_x}(t_q, t_d)}_{\text{temporal relevance}} \tag{4.9}$$

where t_d is the timestamp of document d, f_{s_x} returns the likelihood that instant t_d is relevant to the query q according to the estimation of the temporal relevance of the source s_x. The temporal source s_x can be any of the temporal sources described in the previous section. The coefficients α_i, β_j and γ_x correspond to the feature weights tuned using learning to rank.

The RMTS model (Ranking with Multiple Temporal Signals) is composed of three independent parts that capture different statistics of the information domain. This model is a well-grounded method that generalizes the integration of multiple temporal evidence into a single unified retrieval model. At this point, the divide between previous work and our proposal becomes clear. While previous work relies primarily on the corpus as a single source of temporal signals, the proposed approach looks at additional sources of temporal signals such as users' behavior to improve the estimation of temporal relevance. The factorization of temporal information into a set of rich independent temporal signals provides a more robust way to disambiguate the time periods that are relevant for each search query.

4.3.2 Non-Temporal Features

The non-temporal features in Table 4.1 include scores of text similarity functions computed over the text of the tweets and domain-specific features such as the number of hashtags in the tweet or the number of followers of its author.

The first set of non-temporal features consider retrieval models that have been used as retrieval baselines and have proven to be effective. The BM25 text similarity feature is one of the most popular text retrieval models. The query-likelihood model with Dirichlet smoothing (Zhai and Lafferty, 2004) gives the text similarity score between the query q and the tweet text d.

Since documents in microblogs are very short, term frequency could be less important. The Inverse Document Frequency (IDF) is isolated into a separate text similarity feature even though it is already embedded in BM25. It should allow the learning to rank framework select the optimal coefficient to weight the importance of rare words separately. Intuitively, longer documents have a higher chance of being relevant since they can be more informative, therefore the number of words the text of the tweet is a feature as well.

In addition to text-based features, the model includes the microblog-specific features described in Table 4.1. These features are extracted from textual contents of the tweets' text or users' metrics extracted from the tweets' metadata. This set of features captures information such as number of mentions, number of followers, number of URLs, and other microblog-specific features.

4.3.3 Computing the Model Coefficients

First, we turned our attention to the problem of correctly estimating the temporal density of individual temporal sources. The estimated temporal profile of the different sources is quite heterogeneous, indicating different temporal patterns across the timeline that can be integrated using a feature model.

Second, we addressed the computation of the model coefficients (α_i, β_j, and γ_x) that weight the contribution of each corresponding feature to the final score. Coordinate ascent (Metzler and Croft, 2007) was used to estimate the parameters of the linear model. This learning to rank method is often used to optimize the MAP retrieval accuracy metric directly. Previously, it was used for microblog posts ranking using quality features (Choi *et al.*, 2012) and outperformed alternative learning to rank methods on microblog posts datasets (Xu *et al.*, 2014).

Table 4.1: Non-Temporal Ranking Features

Feature name	Feature description
LM.Dir	Language modeling score for tweet-text.
BM25	Okapi BM25 score for tweet-text.
IDF	Sum of term IDF in tweet-text.
Length	Tweet-text length.
NumURLs	Number of URLs in tweet-text.
HasURLs	1 if tweet-text contains URLs, otherwise 0.
NumHashtags	Number of Hashtags in tweet-text.
HasHashtags	1 if tweet-text contains Hashtags, otherwise 0.
NumMentions	Number of Mentions in tweet-text.
HasMentions	1 if tweet-text contains Mentions, otherwise 0.
isReply	1 if tweet-text is a reply, otherwise 0.
NumStatuses	Number of user's statuses.
NumFollowers	Number of user's followers.

Used by RMTS method and other methods based on learning to rank.

Table 4.2: Temporal Ranking Features

Feature name	Feature description
Recency (**R**)	Recency prior (Li and Croft, 2003).
Twitter Feedback (**TF**)	Temporal feedback (Efron *et al.*, 2014).
Wikipedia Views (**WV**)	Wikipedia article page views.
Wikipedia Edits (**WE**)	Wikipedia article page edits.
News	News headlines.

All temporal features used in RMTS (except for Recency) are produced using kernel density estimation of time series extracted from: the initially retrieved timeline, Wikipedia and News.

4.3.4 Query-Dependent Ranking

The formalization assumes that all sources of temporal signals are relevant to all queries, (i.e., the γ_x coefficient is constant for every query). However, a more in-depth inspection of the temporal profiles of the queries revealed that some queries exhibit stronger temporal patterns than other queries. Queries were grouped into two categories: temporal and atemporal. This distinction

allowed the training of two models: a temporal model that includes all temporal features, and an atemporal model that uses only non-temporal features. So, at query time, the system can now decide to use either the RMTS model (Eq. (4.9)) for temporal queries or the LTR model (Eq. (4.8)) for atemporal queries. We assume that a query is temporal if when using it to retrieve from the news source any news document is returned. Otherwise, the query is classified as atemporal.

Table 4.3: Temporal Ranking Methods Results

	Temporal		Atemporal		T+A		All	
Method	MAP	P30	MAP	P30	MAP	P30	MAP	P30
Text retrieval baselines and LTR								
BM25	0.4054	0.6202	0.4319	0.6392	0.4136	0.6261	0.4136	0.6261
IDF	0.4275	0.6561	0.4360	0.6235	0.4301	0.6461	0.4301	0.6461
LM.Dir	0.4331	0.6491	0.4112	0.6020	0.4264	0.6345	0.4264	0.6345
LTR	0.4688	0.6991	0.4308	0.6216	0.4571	0.6751	0.4528	0.6703
Temporal ranking baselines and RMTS								
Recency	0.4429	0.6667	0.4152	0.6196	0.4343	0.6521	0.4297	0.6552
KDE(score)	0.4621	0.6711	0.4030	0.5961	0.4438	0.6479	0.4455	0.6509
RMTS	$0.5011^{\ddagger *}$	$0.7254^{\ddagger}$	0.4422	0.6353	$\mathbf{0.4829}^{\ddagger *}$	$0.6976^{\ddagger *}$	$\mathbf{0.4809}^{\ddagger *}$	$0.6939^{\ddagger *}$

Symbols † and * stand for a $p < 0.05$ statistical significant improvement over KDE(score) and LTR respectively (‡ and * for $p < 0.01$).

4.4 Evaluation

This section presents the evaluation of the methods on the TREC microblog search test-bed. The method proposed is benchmarked against current state-of-the-art methods. In the TREC microblog *ad hoc* search task, the user wishes to find the most recent and relevant posts. The task can be summarized as: at time t, find *tweets* about topic q. Therefore, systems should favor highly informative *tweets*, relevant to the query, that were published before the query time. These experiments approach time-aware retrieval as a re-ranking problem

given an initial list of *tweets* retrieved using a standard retrieval method (i.e., query-likelihood model).

4.4.1 Datasets and Protocol

TREC datasets. The experimental dataset chosen was the Tweets2013 corpus and the topics for the 2013 and 2014 editions of the TREC Microblog track. The Tweets2013 corpus is much larger (240 million *tweets*) than the Tweets2011 corpus (16 million *tweets*) used in the 2011 and 2012 editions. This collection of *tweets* was created by crawling Twitter's public stream sample via the Twitter streaming API over the period spanning from 1 February 2013 – 31 March 2013 (inclusive). NIST provided the relevance judgments for the 60 topics in TREC 2013 and the 55 topics in the TREC 2014. Judgments were made on a three-point scale of "informativeness": not relevant, relevant, and highly relevant.

Filtering Duplicates and Languages. *Retweets* were considered not relevant as they were seen as duplicates according to the TREC Microblog track evaluation design. Therefore, *Twitter-style retweets* were filtered by looking at the metadata, and *RT-style retweets* were filtered by removing search results that start with *RT*.

In addition, since NIST assessed only *tweets* written in English, *tweets* written in other languages were removed. This language filter is based on the language detection library ldig[4] with a trained model for 19 languages.

Sources. The sources of temporal evidence are the corpus (Twitter), Wikipedia article page views, Wikipedia article edits, and the news sources (USA Today, CNN, and Associated Press).

Protocol. To allow the comparability to previous work, the TREC 2013 topics were used for training the models and the TREC 2014 topics were used for testing. The training data was split into 80% for training and 20% were used for validation. The model was trained by optimizing mean average precision (MAP) using the coordinate ascent learning to rank algorithm.

[4] http://github.com/shuyo/ldig

Following the protocol of the TREC Microblog track, the retrieval accuracy was measured using MAP and P30. The statistical significance of the differences in effectiveness metrics was determined using two-sided paired t-tests following the recommendations by Sakai (2014).

4.4.2 Methods

Firstly, we use as baselines three standard text retrieval methods, BM25, IDF, and LM.Dir. In addition, we evaluate two time-aware ranking methods, Recency and the state-of-the-art KDE(score) (see Table 4.4) for details.

Table 4.4: Methods

Method	Method description
BM25 (Robertson *et al.*, 1994)	Parameterized with $k_1 = 1.2$ and $b = 0.75$.
IDF	No TF weighting might help the retrieval of *tweets*.
LM.Dir (Zhai and Lafferty, 2004)	QL retrieval model with Dirichlet smoothing.
Recency (Li and Croft, 2003)	Time-based language models baseline.
KDE(score) (Efron *et al.*, 2014)	Temporal Feedback.
LTR (Table 4.1)	Learning to rank with non-temporal features.
RMTS	Ranking with Multiple Temporal Sources.

Relevant parameters tuned using learning to rank.

4.4.3 Experimental Results

The methods were evaluated using the setup described and the results are presented in Table 4.3. Special care was taken so that no method used future evidence, and all methods rely only on information available at query time.

Retrieval performance over all queries. Firstly, we evaluated the performance of standard text retrieval methods over the full set of queries (All). The IDF retrieval function outperformed the other text retrieval baselines in both metrics, MAP and P30. The intuition for this result is that higher term frequency (TF) might not correlate with higher relevance in short texts. Therefore, this explains why both BM25 and LM.Dir had a lower retrieval accuracy. These functions put a

stronger weight on term frequency for ranking microblogs. Over-weighting term frequency can also have other adverse effects in microblogs, such as ranking spam results higher due to word repetition.

Secondly, the two temporal ranking baselines that use only the collection as a single source of temporal relevance evidence outperformed all the text retrieval baselines. Of the two, KDE(score) was better and outperformed Recency in MAP, 0.4455 and 0.4297 respectively.

The proposed method RMTS, which uses a learning to rank framework to combine temporal evidence of the collection with several additional sources of temporal signals from the Web, obtained the top results in MAP and P30 outperforming the state-of-the-art temporal ranking methods. Table 4.3 is annotated with symbols denoting the statistical significance level ($p < 0.01$ or $p < 0.05$) of differences in effectiveness to the LTR and KDE(score) methods.

RMTS produced statistically significant differences for MAP and P30. MAP improved 0.0281 over LTR, which corresponds to a relative improvement of 6.2%. We follow the recommendations of Sakai (2014) to report the results and statistical significance tests. According to a two-sided paired t-test for the difference in MAP $\bar{d} = 0.0281$ (with the unbiased estimate of the population variance $V = 0.0020$), RMTS statistically significantly outperforms the LTR model ($t(54) = 4.67$, $p < 0.000020$, $ES_{pairedt} = 0.64$, 95% CI $[0.0161, 0.0400]$). Additionally, according to a two-sided paired t-test for the difference in MAP $\bar{d} = 0.0355$ (with the unbiased estimate of the population variance $V = 0.0049$), RMTS statistically significantly outperforms KDE(score) ($t(54) = 3.73$, $p < 0.000468$, $ES_{pairedt} = 0.51$, 95% CI $[0.0165, 0.0544]$).

Surprisingly, P30 achieved a very competitive result, which is in the same range as the top submitted runs in TREC Microblog 2014. Considering that the top TREC systems use techniques such as pseudo-relevance feedback (PRF), this result becomes an important takeaway message: temporal signals provide key information for ranking microblog search results. According to a two-sided paired t-test for the difference in P30 means $\bar{d} = 0.0236$ (with the unbiased estimate of the population variance $V = 0.0049$), RMTS statistically significantly outperforms LTR ($t(54) = 2.48$, $p < 0.0164$, $ES_{pairedt} = 0.34$, 95% CI

[0.0047, 0.0426]). Additionally, according to a two-sided paired t-test for the difference in P30 means $\bar{d} = 0.0430$ (with the unbiased estimate of the population variance $V = 0.0117$), RMTS statistically significantly outperforms KDE(score) ($t(54) = 2.92$, $p < 0.005126$, $ES_{pairedt} = 0.40$, 95% CI [0.0137, 0.0723]).

Temporal Query Performance Prediction. TREC datasets contain both temporal and atemporal queries, hence it should be advantageous to classify the query to decide on the use of the temporal signals or not. Ranking atemporal queries using temporal features could lead to performance degradation and lower retrieval effectiveness. Despite the small number (60) of training queries the TREC 2013 queries were split into the two classes using a simple strategy: 32 queries that have matching news articles are classified as temporal and 28 queries that do not match news articles are classified as atemporal. TREC 2014 queries, used for testing, were split into 38 temporal queries and 17 atemporal queries – results are shown in Table 4.3.

Results showed that training and ranking using two different models (T+A column in Table 4.3) slightly outperformed RMTS trained using all queries. In the atemporal set of queries, Recency (Li and Croft, 2003) and KDE(score) (Efron *et al.*, 2014) had poorer results for atemporal queries, since these models assume that all queries are time-sensitive, and rely exclusively on a single source of temporal evidence. Thus, temporal query classification is a promising, yet difficult, research direction that can further improve time-aware ranking models.

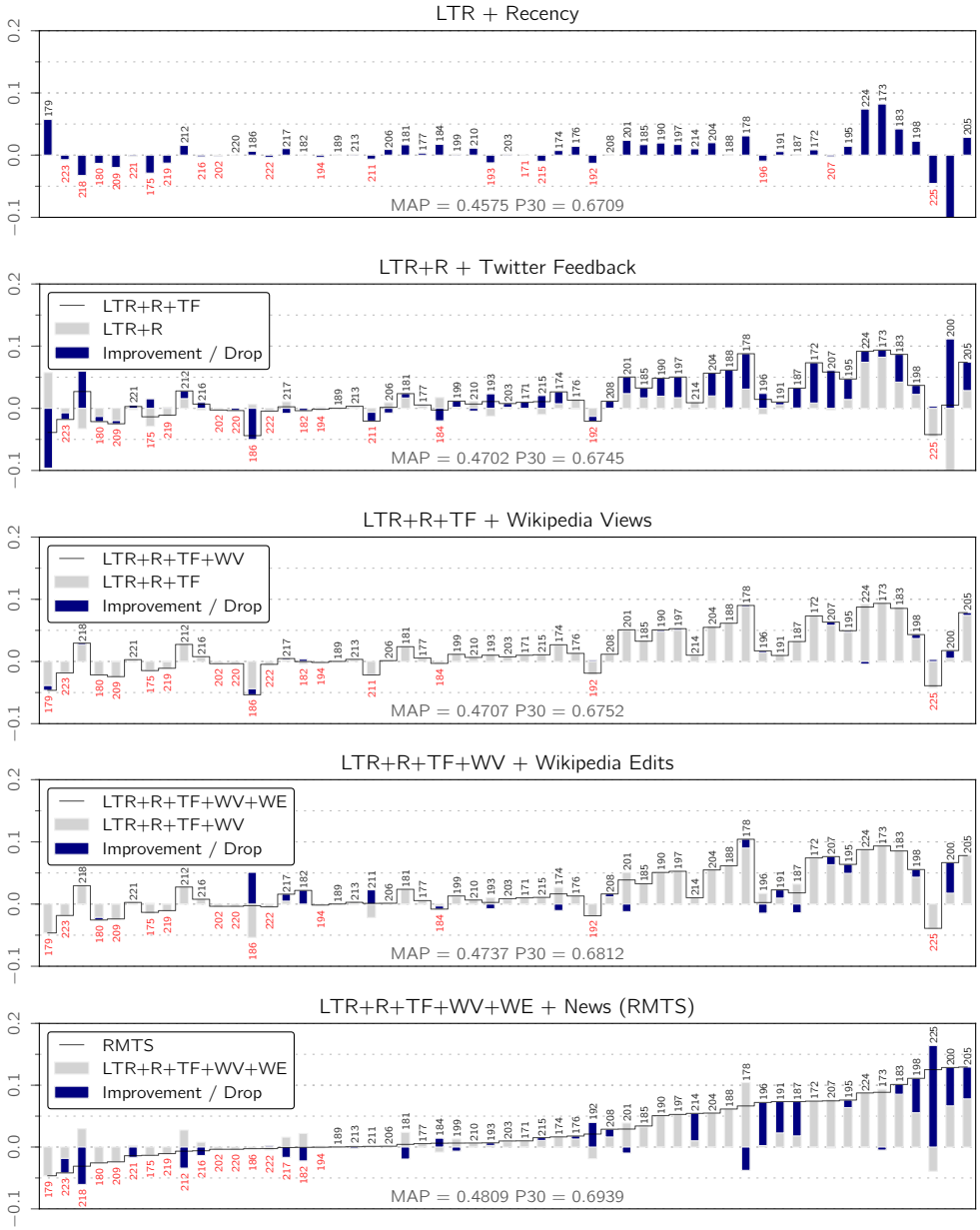


Figure 4.2: Per-feature retrieval results of the RMTS model: graphs show AP relative improvements over LTR by adding each temporal feature incrementally to the LTR model. Each graph illustrates the per-query results, where bars are labeled with the TREC topic number. Topic labels appear above/below if the performance improved/dropped relative to LTR.

4.5 Discussion

In this section results are discussed in three parts. First, a per-feature and per-query analysis of the RMTS ranking model. Second, we examine the contributions of the individual sources to the final ranking. Finally, we examine the robustness of RMTS to missing sources.

4.5.1 Per-Feature and Per-Query Analysis

In general, temporal ranking methods tend to improve overall system performance at the cost of degrading the performance of a smaller number of queries. In Figure 4.2 we present the per-query differences in Avg. Prec. relative to the LTR system to analyze the cumulative gains in performance of each feature across the set of test queries. The proposed RMTS method can improve over LTR in most queries (36 out of 55). Only a few queries had degraded accuracy when using RMTS. This result could be an indication that some sources were incorrectly selected or that the temporal information therein was of low quality. This problem has been widely recognized in temporal query performance prediction work. In these experiments, we decided not to filter or select temporal information sources. Instead, temporal signals from all sources are used for all queries to understand their role in the final retrieval model better.

The use of *recency priors* improved the Avg. Prec. of some queries, however since not all temporal queries favor recency, the accuracy of results was hurt with this method for other queries. It became clear that some queries in this dataset are not temporal: the *recency priors* and the temporal feedback models could not provide more accurate results than LTR in 29 and 26 queries respectively (from a total of 55 test queries).

The proposed model, RMTS, could not improve on 17 queries. The difference is only ~10% of the total number of queries. A possible interpretation is that some of these queries are not temporal. This hypothesis can be confirmed by looking at the failing query topics. For instance, Topic MB223 (“dog off leash”) seems to be an atemporal query, which should not improve with temporal ranking.

A deeper look into this topic showed that every temporal source degraded its performance and the Avg. Prec. dropped almost 0.05. Of note, queries MB179 ("care of Iditarod dogs") that improved with the Recency feature and Topic MB212 ("Kate Middleton maternity wear") that improved with Recency and Twitter Feedback, but the gains obtained by ranking with internal information ended up getting reversed when considering external sources.

In the third graph, it appears that the Wikipedia Views feature does not contribute a lot to the improvements in accuracy. However, the weight learned for this feature using learning to rank was lower than the others in the complete RMTS model. However, when a separate model is trained using only the Wikipedia Views as a temporal feature, the results improve, (see Table 4.5), which could mean that the Wikipedia feature was redundant for some queries in the complete model. Moreover, while P30 improved with the introduction of the WV feature (see Table 4.6), withholding it improved MAP results slightly.

Finally, not all queries used the full range of external sources. Some queries had no matching news articles or matched Wikipedia articles that did not provide useful evidence for the time-span of the corpus.

4.5.2 Contributions of Individual Sources

The use of multiple temporal signals calls for a more in-depth analysis of the contributions that each source makes to the final rank. In this experiment, we examine the contribution of each temporal source to the improvement in MAP over the LTR model. The results in Table 4.5, show that the relative improvements in MAP with each individual temporal signal are in the range of +2.67% to +4.51%, reaching +6.02% with RMTS. The most balanced MAP and P30 improvements were obtained using the News sources and Wikipedia edits.

The best improvement in MAP is obtained with the Twitter Feedback feature and the best improvement in P30 is obtained by the Wikipedia Edits feature. However, no source was particularly more effective than the others across both retrieval metrics as evidenced by the results obtained.

In summary, all temporal signals f_{s_x} had a positive impact in the accuracy of

the results, but when combined as a whole using a learning to rank framework, they outperformed all other models with a MAP of 0.4809 and P30 of 0.6939. Actually, this is in line with our hypothesis: each temporal information source f_{s_x} can provide different temporal signals or temporal patterns useful to estimate the relevant time periods for a query.

Table 4.5: Contributions of Individual Temporal Sources.

Method	MAP	P30
LTR	0.4528	0.6703
LTR_{+R}	$+2.67\%^{*}$	+1.36%
LTR_{+TF}	$+4.51\%^{\ddagger}$	+0.54%
LTR_{+WV}	$+2.67\%^{*}$	+1.00%
LTR_{+WE}	$+2.78\%^{*}$	+2.16%
LTR_{+News}	$+3.05\%^{\ddagger}$	+1.72%
RMTS	$+6.21\%^{\ddagger}$	$+3.52\%^{*}$

Figures are relative improvement over non-temporal baseline. Symbol * stands for a $p < 0.05$ statistical significant improvement over LTR (‡ for $p < 0.01$).

Table 4.6: Ranking Robustness to Missing Sources.

Method	MAP	P30
RMTS	0.4809	0.6939
RMTS_{-R}	$-0.94\%^{*}$	$-1.66\%^{*}$
RMTS_{-TF}	−1.31%	−1.12%
RMTS_{-WV}	+0.48%	−0.95%
RMTS_{-WE}	$-1.10\%^{*}$	−0.76%
RMTS_{-News}	$-2.74\%^{*}$	$-1.90\%^{*}$

Symbol * stands for a $p < 0.05$ statistical significant difference compared to RMTS (‡ for $p < 0.01$).

4.5.3 Robustness to Missing Sources

To further examine the robustness of the proposed method we studied the impact that a missing temporal source would cause in the final rank. This is a quite practical aspect in a real production system, where a temporal source might be temporarily unavailable. Table 4.6 presents the MAP and P30 results. Two key facts arise from this table: the first one is related to the influence of the News temporal signals, and the second one concerns the low drop in MAP performance caused by missing any other temporal source.

In general, MAP results are only slightly affected by each individual missing source – in the worst case, MAP drops 2.74% when withholding the News. We consider this to be an excellent measure of the robustness to missing temporal sources. Moreover, this further hints that multiple sources provide complementary temporal signals. On the one hand, the most important discovery is that the News is a key temporal information source of temporal signals. On the other hand, the News feature can be considered as a group of news sources and not a single news source. Thus, the drop in the number of temporal information sources with its removal could be considered disproportionate compared with the other features.

4.6 Summary

This chapter presented the RMTS framework that mines the behavior of the crowds for temporal signals. This novel time-aware ranking method integrates lexical, domain, and temporal evidences from multiple Web sources to rank microblog posts. It explores the signals from Wikipedia (through page views and page edit history), news articles, and Twitter feedback to estimate the temporal relevance of search topics.

Retrieval precision. We evaluated our system following the experimental setup of the microblog evaluation track at TREC 2013 and TREC 2014. The results of the experiments confirmed our hypothesis that multiple external sources of temporal signals exhibit temporal patterns that can be used effectively for ranking

microblog results. The proposed method statistically significantly outperformed BM25 and the Language Model retrieval model with Dirichlet smoothing by 13.2%. It also statistically significantly outperformed a strong baseline based on learning to rank that uses several lexical and domain features.

RMTS is less biased. A key advantage of the proposed RMTS model is its robustness and stability: the improvement over the learning to rank model with non-temporal features could not be pinpointed to a single source of temporal evidence. Moreover, this approach is more resilient to missing temporal information sources.

Unified representation of temporal signals. The proposed framework offers a principled methodology for mining and representing temporal signals from multiple temporal information sources. It allows predicting the temporal relevance of queries from heterogeneous pairs of timestamps and weights mined from diverse temporal information sources.

Effective use of Wikipedia temporal signals. The behavior dynamics of Wikipedia users is an adequate source of temporal evidence for time-aware ranking. Previous works exploit Wikipedia for detecting events and linking entities related to the events using either page views statistics (Ciglan and Nørvåg, 2010) or page revision history (Georgescu *et al.*, 2013; Steiner *et al.*, 2013). However, to the best of our knowledge, this is the first work that exploits multiple external sources for time-aware ranking.

Since news sources are a more accessible resource to crawl, the next steps could delve into the scaling of the number of news sources crawled to obtain better coverage of topics. However, new research questions arise: for a query q *which news sources should be selected?*, and *how to weight them for each query?*.

The following chapter is concerned with the combination of the approaches described in the two previous chapters into a ranking framework that uses time-aware pseudo-relevance feedback and temporal evidence from external collections. The proposed method is the main contribution in this thesis.

MODELING TEMPORAL EVIDENCE FROM EXTERNAL COLLECTIONS

5

"They desire to look at relevant results from important subtopics from the most relevant time points of interest."

— Singh *et al.*, 2016

Newsworthy events are broadcast through multiple mediums and prompt crowds to produce comments on social media. This chapter proposes a method that leverages this behavioral dynamics to estimate the most relevant time periods for an event (i.e., query topic). Recent advances have shown how to improve the estimation of the temporal relevance of such topics. In this approach, we build on two major novelties. First, to improve the robustness of the detection of relevant periods, we mine temporal evidence from *hundreds of external sources* that are aggregated into topic-based external collections. Second, we detail a formal retrieval model that *generalizes the use of the temporal dimension* across all aspects of the retrieval steps.

In particular, we show how the temporal dimension is used to (i) infer a topic's temporal evidence from different crowds, (ii) select the query expansion terms, and (iii) re-rank the final results for improved precision. Experiments with TREC Microblog collections show that the proposed temporal retrieval model makes effective and extensive use of the temporal dimension to improve search results over the most recent temporal models. Interestingly, we observe a strong temporal correlation between retrieval precision and the distribution of retrieved and relevant documents.

A networked world and the increasing pervasiveness of Internet access enables the rapid adoption of new online communication mediums to discuss current events. Previous research has explored this symbiosis between Twitter and the news (Kwak *et al.*, 2010; Sankaranarayanan *et al.*, 2009) and link the

two mediums (Guo *et al.*, 2013; Tsagkias *et al.*, 2011). Events are discussed on the Web as they happen and the people following them can add to the conversation on current topics immediately. Hence, improving *temporal relevance estimation* for searching such events became a significant research priority.

The now standard Web search retrieval schemes based on learning-to-rank feature models and retrieval functions based on language modeling have proven to be very effective. However, relevance on Twitter has many dimensions: authority, popularity, freshness, geographical context, and topical relevance.

Previously, time-aware ranking research explored the assumption that fresh documents are more relevant (Li and Croft, 2003). Later models revised this assumption in line with what is observed in Twitter: for time-sensitive queries, documents tend to cluster temporally (Dakka *et al.*, 2012; Efron *et al.*, 2014). Our approach is based on the intuition that discussions about an event are likely to occur around the same time periods across multiple mediums and sub-topics.

The rationale is that newsworthy events trigger a cascade of activity on the Web and Twitter. This information can be useful for ranking and, in some cases, can be gathered with ease. The news often have a good coverage of current topics, clean journalistic language, and reliable timestamps. Thus, it is desirable to aggregate news sources to offer more context to the *tweets* as well as to the users' queries intent. In particular, we aim to explore the *crowd aggregation effect* to extract temporal evidence from news verticals. Temporal evidence is further used to boost the selection of query expansion terms and to refine query topics temporal relevance. This approach is completed with the re-ranking of the final search results leading to improved precision. Hence, the proposed method brings a series of novel contributions:

- Explore the crowd effect by aggregating news sources into verticals;
- Mining of crowds' temporal evidence at different granularities (i.e., *verticals*, *documents* and *terms*);
- A formal time-aware ranking model that unifies multiple temporal features into a single, but comprehensive, temporal retrieval model.

Including the temporal dimension at the different steps of the search engine pipeline, improves the accuracy of several small tasks, leading to greater overall gains. The temporal dimension introduces stronger evidence in many decision tasks (e.g., selection of query expansion terms). Evaluation on the TREC 2013 and TREC 2014 Microblog Track datasets shows that the proposed retrieval model outperforms state-of-the-art methods.

In contrast to previous work, we propose to use multiple news verticals to robustly identify the relevant time periods for each query, instead of relying only on the temporal distribution of pseudo-relevant documents (Dakka *et al.*, 2012) or first-order statistics from verticals (Arguello *et al.*, 2011). This chapter is organized as follows: in Section 5.1 the formal temporal ranking model is detailed and the following sections detail its implementation; evaluation is presented in Section 5.3; and a more fine-grained discussion in Section 5.4.

5.1 Modeling Temporal Evidence

Consider a corpus containing N documents, represented by D. To integrate the temporal relevance component in the ranking model Dakka *et al.* (2012) decomposed the document in two different parts: lexical evidence, the words in the document (w_d), and temporal evidence, the document's timestamp (t_d).

We consider an augmented ranking model that contemplates query-independent signals or metadata from the document, m_d, in addition to the lexical and temporal evidence as follows:

$$\begin{aligned} P(d \mid q) &= P(w_d, t_d, m_d \mid q) \\ &\propto P(w_d \mid q) \cdot P(t_d \mid q) \cdot P(m_d \mid q) \\ &\propto \underbrace{P(q \mid w_d) \cdot P(w_d)}_{\text{query-likelihood model}} \cdot P(t_d \mid q) \cdot P(m_d) \end{aligned} \tag{5.1}$$

where the final formulation follows from the two following steps: First, by applying the Bayes' rule to $P(d \mid q)$ and eliminating the quotient $P(q)$ based on the

rank equivalence to get the well-known query-likelihood retrieval model. Second, by assuming the independence between document metadata and the query, $P(m_d)$ can be taken as the query-independent importance of the document.

To instantiate the ranking model from Eq. (5.1), we need to estimate three components: lexical, temporal, and query-independent. The lexical component can be estimated using *relevance models* (see Subsection 5.1.2) or standard query-likelihood, where as usual we assume that $P(w_d)$ is uniform.

In this chapter, we focus on estimating the temporal component using external collections (see Subsection 5.1.1). The query-independent component can be estimated using values extracted from the metadata of the document (see Table 5.1). To estimate the temporal component, former models (Dakka *et al.*, 2012; Efron *et al.*, 2014) assume that relevant temporal information is only available on the search corpus itself, D, for instance via the temporal distribution of an initial set of feedback documents. However, temporal feedback on the corpus alone can be boosted by external sources (Martins *et al.*, 2016a). Therefore, we propose estimating temporal relevance using external collections:

$$\begin{aligned} P(t_d \mid q) &= P(t_d \mid q, D, \mathscr{C}) \\ &= P(t_d \mid q, D) \cdot P(t_d \mid q, \mathscr{C}), \end{aligned} \tag{5.2}$$

where the last step follows if we assume that temporal evidence can be extracted from the search corpus D and from the external collections $\mathscr{C} = \{c_1 c_2 \cdots c_{|\mathscr{C}|}\}$ independently. The first part can be estimated using the temporal distribution of feedback documents retrieved using the query q, (Efron *et al.*, 2014). We calculate the temporal relevance according to the external collections as described in Subsection 5.1.1.

We also propose to generate query expansions to improve the document ranking (i.e., the lexical component) by leveraging the external collections to estimate time-based *relevance models*. In Subsection 5.1.2, we present a novel external time-based relevance model to generate expanded query models for retrieval in the corpus. The expanded query model is computed by taking into account lexical as well as temporal evidence contained in the collections.

5.1.1 External Temporal Relevance

For a given query, different collections yield different temporal relevance estimates (i.e., different probability distributions of relevance over time). Therefore, we need to extend Eq. (5.2) to combine all the different temporal relevance estimates from each external collection into a single robust estimate. In our approach, we combine them using a weighted mixture of probability distributions

$$\begin{aligned} P(t_d \mid q, \mathscr{C}) &\propto \sum_{c \in \mathscr{C}} P(t_d \mid q, c) \cdot P(c \mid q) \\ &\propto \sum_{c \in \mathscr{C}} P(t_d \mid q, c) \cdot P(q \mid c) \cdot P(c), \end{aligned} \tag{5.3}$$

where $P(t_d \mid q, c)$ is the importance of time t_d for the query q in the collection c, $P(q \mid c)$ is the relevance of the collection c to the query q, and $P(c)$ is the query-independent collection prior.

Considering that we may have several collections, the calculation of temporal relevance over all of them raises efficiency concerns. Therefore, we follow a *federated search* approach (Shokouhi and Si, 2011), and consider that only a few collections contain most of the temporal evidence for a given query q. Therefore, we can use just those to provide an adequate approximation

$$P(t_d \mid q, \mathscr{C}) \propto \sum_{c \in \mathscr{C}_q} P(t_d \mid q, c) \cdot P(q \mid c) \cdot P(c), \tag{5.4}$$

where $\mathscr{C}_q$ is a ranking of the most relevant collections to query q, and the query-independent prior of the collection is considered uniform $P(c) = 1/|\mathscr{C}_q|$

Considering M_k, the final single ranking obtained by merging all the results retrieved from the selected collections $\mathscr{C}_q$, the relevance of collection c is given by the ratio between the number of its documents that make it into the top ranking, M_c, by the total documents retrieved, M_k:

$$P(q \mid c) = \frac{|M_c|}{|M_k|} \tag{5.5}$$

We opted for an approach that is similar to resource selection algorithms, such as ReDDE (Si and Callan, 2003). There are several options to estimate $P(q \mid c)$, including other resource selection algorithms (Aly *et al.*, 2013; Kulkarni *et al.*, 2012; Shokouhi, 2007; Si and Callan, 2003; Weerkamp *et al.*, 2012).

5.1.1.1 Vertical Temporal Feedback

For each document d we would like to find $P(t_d \mid q, c)$, the probability of relevance of its timestamp t_d according to vertical c and the query q. This probability follows the joint distribution $f_c(t_d)$,

$$P(t_d \mid q, c) \sim f_c(t_d). \tag{5.6}$$

The probability density function $f_c(t_d)$ is estimated over the time span of the collection using a weighted kernel density estimation method:

$$f_c(t) = \frac{1}{nh} \sum_{d \in \mathcal{R}_c} \lambda_d \; K\left(\frac{t - t_d}{h}\right) \tag{5.7}$$

where t is the timetamp of the input document, $\mathcal{R}_c$ is the set of retrieved documents from the collection c and t_d corresponds to these documents' timestamps. The kernel function $K(z)$ corresponds to the Gaussian kernel $\mathcal{N}(z, 0)$, and the optimal bandwidth can be estimated using Silverman's rule-of-thumb $h^* \approxeq 1.06\,\sigma\, n^{-1/5}$. Finally, λ_d, is a non-negative weight on timestamp t_d, to weight each timestamp by its importance.

5.1.2 External Time-based Relevance Models

Relevance models provide a framework for term selection and estimation of the importance of terms for query expansion (Lavrenko and Croft, 2001). We propose to estimate relevance models and generate a final expanded query q' using external collections $\mathcal{C}$, leveraging their temporal evidence,

$$P(q \mid w_d, \mathcal{C}) \approx P(q' \mid w_d). \tag{5.8}$$

Let θ_q be the original query model and $\theta_{F_{\mathscr{C}}}$ an estimated feedback query model based on feedback documents $d_1 \cdots d_k$ from multiple external collections. Let the final query model be $\theta_{q'} = (1-\alpha)\ \theta_q + \alpha\ \theta_{F_{\mathscr{C}}}$ (Zhai and Lafferty, 2001). The final query is then a linear combination of the original query model, $P(w \mid \theta_q)$, and the feedback query model, $P(w \mid \theta_{F_{\mathscr{C}}})$, using external collections:

$$P(w \mid \theta_{q'}) = \lambda \cdot P(w \mid \theta_q) + (1-\lambda) \cdot P(w \mid \theta_{F_{\mathscr{C}}}), \tag{5.9}$$

where the original query is modeled using its maximum-likelihood estimate $P(w \mid \theta_q) = \#(w,q)/|q|$. Time is introduced in the second parcel of the above expression to improve the estimation of the feedback query expansion terms.

To this end, we integrate temporal feedback into term selection. We introduce a novel generative model of the query that first selects a collection, then a date, and then selects a term based on the collection, date and query.

$$P(w \mid \theta_F) = \sum_{c \in \mathscr{C}} \sum_{T} P(w \mid T, q, c) \cdot P(c \mid T, q) \cdot P(T \mid q) \tag{5.10}$$

$$\propto \sum_{c \in \mathscr{C}} P(c \mid q) \sum_{T} P(w \mid T, q, c) \cdot P(T \mid q) \tag{5.11}$$

where $P(w \mid T, q, c)$ is the importance of the word w in day T for the query q given collection c, $P(c \mid T, q)$ is the importance of collection c in day T for the query q, and $P(T \mid q)$ is the importance of the day T to the q.

$$\propto \sum_{c \in \mathscr{C}} P(c \mid q) \sum_{T} P(T \mid q) \sum_{d \in \mathscr{R}_T} P(w \mid d) \cdot P(q \mid d) \cdot P(d) \tag{5.12}$$

$$\propto \sum_{c \in \mathscr{C}} P(c \mid q)\ \frac{1}{|\mathscr{R}_c|} \sum_{d \in \mathscr{R}_c} P(w \mid d) \cdot P(q \mid d) \cdot P(t_d \mid q) \tag{5.13}$$

where $\mathscr{R}_T$ is the number of retrieved documents on a given day T. The first step converts the time variable from the discrete domain (days), similarly to Choi and Croft (2012), to the continuous domain (document timestamps). Hence, in this formulation $P(t_d \mid q)$ can be estimated for document timestamps t_d. Instead of estimating the temporal relevance over the retrieved timeline by generating

temporal slices we use kernel density estimation (Efron *et al.*, 2014) to estimate a probability density function that provides a smooth estimate of $P(t_d \mid q)$. Finally, we approximate the above expression by selecting the collections $\mathscr{C}_q$ that contribute the most to the final estimation, i.e.,

$$P(w \mid \theta_F) \approx \sum_{c \in \mathscr{C}_q} P(q \mid c) \cdot P(c) \cdot \frac{1}{|\mathscr{R}_c|} \sum_{d \in \mathscr{R}_c} P(w \mid d) \cdot P(q \mid d) \cdot P(t_d \mid q). \tag{5.14}$$

In practice, we assume $P(c)$ to be constant and as in Eq. (5.5), $P(q \mid c) = \frac{|M_c|}{|M_k|}$.

5.2 Learning to Rank External Temporal Evidence

We are now ready to plug-in the temporal evidence and the time-based relevance models from multiple verticals into a common ranking model. To combine the different temporal features extracted from query-specific verticals, we can re-write Eq. (5.1) as the log-linear model

$$\log P(d \mid q) \propto Z + \log P(q \mid w_d) + \log P(t_d \mid q) + \log P(m_d) \tag{5.15}$$

where we can replace $P(t_d \mid q)$ by the temporal relevance over D and $\mathscr{C}$, and the query q by the expanded query q'. Now, using a learning to rank approach to weight the different components of the initial model, we have

$$\log P(d \mid q) \propto Z + \sum_i \alpha_i \log P_i(q' \mid w_d) \tag{5.16}$$

$$+ \beta \log P(t_d \mid q', D) \tag{5.17}$$

$$+ \gamma \log \sum_{c \in \mathscr{C}_q} P(t_d \mid q, c) \cdot P(q \mid c) \tag{5.18}$$

$$+ \sum_j \delta_j \log P(m_d^j), \tag{5.19}$$

where $P(q' \mid w_d)$ is the retrieval score of the document, d, given the expanded query, q'. Instead of using a single estimate of the lexical component, $P(q' \mid w_d)$, more accurate results are obtained by averaging multiple estimates provided by different retrieval models (Eq. (5.16)). The weights α_i indicate the confidence in each retrieval model's estimate. Since query expansion is used for temporal feedback (see Eq. (5.17)) we retrieve using the expanded query.

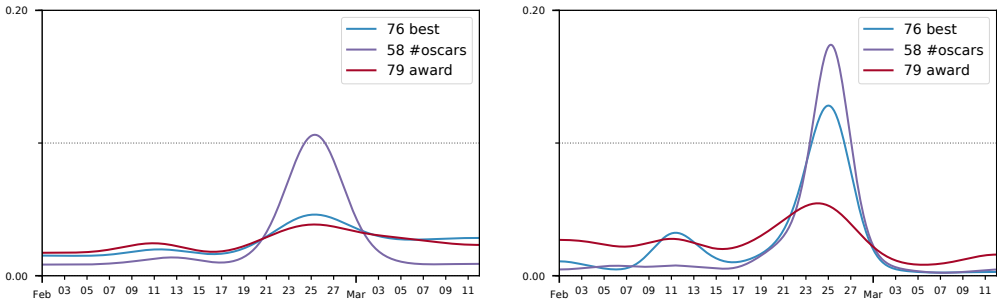
This additional temporal feedback feature according to the corpus, D, itself, is added on top of the estimation of temporal relevance from the external collections introduced by Eq. (5.18). To account for different ways to estimate the importance of a document from metadata we introduce Eq. (5.19). Next, we discuss the relationship between each feature of the above ranking model and the formal model.

Table 5.1: Learning to rank features.

Feature name	Feature description
Doclen	Document length.
#URL	URL count.
#hashtags	Hashtags count.
#mentions	Mentions count.
hasURL	1 if it contains URL, otherwise 0.
hasHashtags	1 if it contains Hashtags, otherwise 0.
hasMentions	1 if it contains Mentions, otherwise 0.
isReply	1 if it is a Reply, otherwise 0.
#statuses	Total number of posts.
#followers	Total number of followers.

Used in the learning to rank methods, including KDE+KDE_E+RMT_E.

Learning to rank features. The proposed model is composed of four main components that capture different aspects of search relevance in time-sensitive collections. First, we employ three different retrieval models to obtain textual matching scores, Eq. (5.16). They are, the query-likelihood retrieval model with Dirichlet prior smoothing (LM.Dir), BM25, and IDF. Second, Eq. (5.17) includes a temporal feedback feature (Efron *et al.*, 2014), calculated over the documents retrieved from the main corpus D with the expanded query q'.



(a) $P(t \mid c)$. (b) $P(t \mid q, c)$.

Figure 5.1: Temporal profiles of queries and collections.

Third, the proposed model generalizes the integration of temporal evidence from external collections, Eq. (5.18), aggregated into a single score. In Eq. (5.18), the importance of the publishing timestamp of the document t_d according to the external collections is estimated by a summation over the likelihood of each selected verticals. For each vertical, $P(t_d \mid q, c)$ returns the likelihood that an instant represented by t_d is relevant to the query q according to, $\mathcal{R}_c(q)$. The coefficients α_i, β, γ, and δ_j correspond to the feature weights. In contrast to previous work that often relies on a single source of temporal evidence, e.g., corpus, the proposed approach contemplates the use of several external collections. The calculation of the temporal evidence feature over the documents retrieved from query-specific verticals can provide a more robust estimation of the relevant time periods for each query.

Fourth, many non-temporal and query-independent features Eq. (5.19), were added to improve effectiveness further, such as quality features (Choi *et al.*, 2012) and other commonly used features in learning to rank approaches to microblog search (Xu *et al.*, 2014). Table 5.1 lists this set of features. This set of features captures microblog-specific information that is useful for ranking such as, number of statuses, followers, URLs, and hashtags. The number of words in the tweet was added as a feature to boost longer documents.

5.2.1 Example: Temporal Evidence from External Collections

Let's examine an example in light of the described model. Consider the 85th Academy Awards ceremony that took place on February 24 2013, at the Dolby Theatre in Hollywood, Los Angeles. The top award winner, winning the Oscar Award for *Best Picture*, was Argo, a movie starring Ben Affleck. This event sparked multiple processes on the Web, such as the dissemination of news articles about the event, and discussions and commentary on Twitter, Figure 5.1. A person interested in surveying the general commentary and opinions could procure a list of relevant accounts to monitor posts in real-time. However, journalists and other searchers would most likely use search engines to find general information outside one's circle about specific aspects.

We dissected how multiple accounts from news outlets and other verified account on Twitter organized into different topical shards can be used in the search process. Using the TREC Microblog query MB195 - "Argo wins Oscar", we plot two graphs that show the temporal distribution of results. Firstly, in Figure 5.1a we show an estimate of relevant time periods using kernel density estimation over all document timestamps of each shard.

In this example we used a resource selection algorithm (Aly *et al.*, 2013) to select the three most useful shards for the query. All three topical shards selected exhibited a larger probability around the time of the live broadcast. Identified by the word "#oscars", Shard 58, is relatively more bursty than the others. Secondly, since topical shards are too broad, we can fine-tune the estimation of the relevant time periods for a given query by finding a further subset of documents that are related to the query. In Figure 5.1b, we improved the estimation by searching over the topical shards selected and for each one using for estimation the documents retrieved by the query "Argo wins Oscar". Shard 79 identified by the word "award" is less spiky than the others. The key insight from this comparison is that two topical shards have the most useful temporal information. This analysis hints that using multiple external collections (i.e., the topical shards), can be useful to detect when events are *happening* and can be more robust than using a single source of evidence (i.e., the corpus).

5.3 Experimental Methodology

This section presents the evaluation of the methods described in the previous sections on the TREC microblog search test-bed. In the TREC Microblog track problem of retrospective *ad hoc* retrieval, the user wishes to find the most up-to-date and relevant posts. The task can be summarized as: at time t, find *tweets* about topic q. Therefore, systems should favor highly informative *tweets* relevant to the query topic that were published before the query time.

5.3.1 Protocol

Our experiments delve into the problem of re-ranking *tweets* sampled using a standard retrieval method (i.e., query-likelihood model) taking into account temporal crowd signals from different sources. In our experiments, we follow TREC and report the MAP and P30 results. Statistical significance of effectiveness differences are determined using two-sided paired t-tests following Sakai (2014).

Filtering duplicates and languages. In the collection used, *retweets* are considered not relevant because they are seen as duplicate documents. Therefore, we filtered *Twitter-style retweets* using the tweet metadata available, and we also filter out *RT-style retweets*. Moreover, assessors evaluated only relevant *tweets* written in English, therefore we use the language filter ldig[1] to remove *tweets* in other languages.

5.3.2 Datasets

The experiments are done on standard TREC Microblog datasets and using a large collection of posts from Twitter Verified Accounts to build fine-grained topical external collections.

[1] https://github.com/shuyo/ldig

5.3.2.1 TREC Microblog

The Tweets2013 dataset is the most comprehensive evaluation resource for *ad hoc* retrieval on social media to date. The Tweets2013 corpus is much larger (≈240 million *tweets*) than Tweets2011 (16 million *tweets*) used in TREC 2011 and TREC 2012. It was created by crawling Twitter's public sample stream over the period spanning from 1 February 2013 - 31 March 2013. The experiments were performed using both the query topics for the 2013 and 2014 editions of the TREC Microblog track (Lin and Efron, 2013b; Lin *et al.*, 2014). NIST provided relevance judgments TREC 2013 (60) and TREC 2014 (55) on a three-point scale: not relevant, relevant, and highly relevant.

5.3.2.2 External Collections: Twitter Verified Accounts

We crawled the timelines of Twitter's verified users (~205k accounts as of Aug 2016) collecting tweets from the period 1 February – 31 March 2013, which matches the period covered by the Tweets2013 TREC microblog dataset. Twitter's verified accounts belong to news organizations, mass media, and celebrities, so the posts have higher quality than a randomly sampled accounts. The cleaner vocabulary also allows the identification of interesting clusters more easily.

Topping the list of verified users (sorted by number of followers), there are a number of singers, actors, and other celebrities. Additionally, some accounts belong to companies that provide customer support through Twitter. These accounts provide customer support using private messages sent via Twitter Direct Messages (DMs). To be able to send DMs on Twitter, users have to follow each other. Thus, to help remove these two types of unwanted accounts, we extract two additional metrics for each account:

- the average number of tweets per day and
- the ratio between the number of replies and total posts.

To select high quality informative sources we remove accounts that meet the following criteria: $posts/day < 10$ and $\frac{replies}{posts} > {}^1/_3$. Accounts that belong

Table 5.2: Topics extracted using mini-batch k-Means and NKL (K=200).

6	28	29	58	63	127	159
marriag	red	pope	#oscars	pistoriu	bank	korea
gay	carpet	franci	jennif	oscar	cypru	north
same-sex	dress	benedict	oscar	bail	bailout	south
suprem	vettel	cardin	lawrenc	hear	tax	nuclear
court	#oscars	vatican	congratul	murder	#cyprus	korean
support	jessica	xvi	win	girlfriend	euro	test
equal	bull	resign	#twitter140	charg	cypriot	sanction
argument	oscar	conclav	includ	court	crisi	rodman
coupl	gown	new	best	reeva	deposit	threat
clinton	chastain	elect	christoph	#oscarp...	central	denni
hillari	look	church	#oscars2013	steenkamp	rate	africa
#scotus	#grammys	cathol	actress	blade	levi	missil
scout	card	papal	list	runner	eurozon	militari
ban	kidman	#pope	ann	#pistorius	atm	threaten
back	tale	smoke	hathaway	case	reopen	war
divorc	naomi	mass	argo	south	barclay	nuke
defens	lipstick	chapel	kristen	shoot	capit	say
#prop8	walli	sistin	cabin	nike	mortgag	warn
legal	grammi	bergoglio	waltz	face	break	china
justic	webber	rome	stewart	detect	asset	vow

to news media outlets and other mass media organizations, typically produce a high volume of posts daily. Thus, we remove accounts that have a low daily average number of posts (e.g., @katyperry, @justinbieber, etc.). News accounts and broadcasters seldom reply to other users on Twitter, while accounts used by companies to provide customer support have a high ratio of replies (e.g., @XboxSupport, @AppleCare, etc.). Each account's timeline is then classified in terms of written language by sampling their five most recent posts using ldig to remove non-English accounts. A total of 645 accounts were used, totaling approximately 800k *tweets*.

Tweets are tokenized using Twokenize[2], initially published alongside TweetMotif (O'Connor *et al.*, 2010). Preprocessing included removing URLs, email addresses, numbers, times, mentions, and emoticons. The tweets corpus was partitioned using mini-batch k-Means, with the number of clusters empirically

[2] https://github.com/myleott/ark-twokenize-py

set to $K = 200$ since the corpus covers a large period of 2 months. The creation of information verticals in an automated way is crucial for scaling to a larger number of documents and sources, and to provide more fine-grained topics for query expansion. Previous work proposed using k-Means and metrics based on the Kullback-Leibler divergence to partition a corpus of text documents into clusters of topically similar documents. We implemented the symmetrized version of the Kullback-Leibler divergence proposed by Kulkarni and Callan (2015) for selective search that is smoothed with a background model (Ogilvie and Callan, 2001a) to compensate for the unequal sizes of documents and clusters. The implementation was validated on a standard dataset using different metrics for comparison. More information is presented on Appendix A. A selection of the topics discovered using the Negative Kullback-Leibler divergence metric over the tweets dataset are shown on Table 5.2.

5.3.3 Baselines and Experimental Systems

Relevance baselines. The first baseline is the QL retrieval model with Dirichlet prior smoothing (Zhai and Lafferty, 2004) with $\mu = 2500$ (i.e., **LM.Dir**). The second strong baseline, **LTR**, is a learning to rank model combining multiple retrieval models (i.e., LM.Dir, BM25, IDF) and the features in Table 5.1.

Temporal baselines. There are three temporal ranking baselines: **Recency** (Li and Croft, 2003) and **KDE(score)** and **KDE(rank)** (Efron *et al.*, 2014), two different variants of a state-of-the-art temporal feedback method.

Experimental systems. The KDE_E method consists in performing temporal feedback on external collections as described in Subsection 5.1.1.1. The RM_E method uses the external collections, described in the previous section, to expand the initial query before searching the main corpus. The RMT_E experiment system uses time-based term expansion introduced in Subsection 5.1.2. Finally, the KDE+KDE_E+RMT_E experimental system uses both temporal vertical feedback and time-based term expansion. Whenever KDE is used, we opted for the KDE(rank) variant due to its better performance on previous publications. For learning to rank we used *coordinate ascent* to optimize MAP.

5.4 Results and Discussion

In this section, we start by comparing the retrieval results of the different baselines, temporal methods, and experimental systems, and then present a qualitative analysis of the temporal distribution of these systems' results.

5.4.1 Estimating Time-based Relevance Models

In this section, we analyze the influence of time-based relevance models. The organization of the expansion corpus into topic-based verticals makes the query expansion process *temporally focused*. Verticals created by a partitioning algorithm using a topic-based similarity criteria exhibited different temporal profiles. The distribution of documents contained in each topic-based vertical is biased towards the time periods for when the vertical is most relevant. Following the temporal cluster hypothesis, the temporal relevance estimate extracted using the timestamps from the verticals selected was integrated into the retrieval process. In the pseudo-relevance feedback term selection stage it is used to generate *temporally focused* query expansion terms. In Table 5.3 and Table 5.4 we present a comparison of the results of MAP and P30 in the TREC 2013 and 2014 test topics. By estimating the relevance models using the proposed time-sensitive term selection approach (RMT_E), the retrieval effectiveness always improved against the non-temporal method (RM_E). In fact, in TREC 2013 we observe a large effect on P30 of time-sensitive term selection when using the proposed vertical feedback architecture. Overall, we found that time-sensitive term selection is effective when used in a standard pseudo-relevance feedback architecture as well as in the proposed vertical feedback architecture.

5.4.2 Estimating Temporal Relevance

In this section, we analyze the importance of temporal feedback from external collections. The major difference between KDE_E and $KDE{+}KDE_E{+}RMT_E$ is that the former uses the vertical feedback architecture for temporal feedback

Table 5.3: TREC 2013 dataset results.

Method	MAP	P30	Rprec
LM.Dir	0.2629	0.4622	0.3094
Recency	0.2663	0.4611	0.3115
KDE(score)	0.2583	0.4517	0.3004
KDE(rank)	0.2736†	0.4878†	0.3178†
LTR	0.2787	0.4617	0.3193
RM_E	0.2797	0.4528	0.3167
RMT_E	0.2824†	0.4700	0.3233
KDE_E	0.2889‡	**0.5061**‡	**0.3322**‡
KDE+KDE_E+RMT_E	**0.2900**†	0.4850	0.3229

Symbols † and * stand for a $p < 0.05$ statistical significant improvement over KDE(score) and LTR respectively (‡ and * for $p < 0.01$).

only, while the latter uses this architecture for query expansion via a time-aware pseudo-relevant vertical feedback method.

In addition, it uses the estimate of temporal relevance obtained from temporal feedback on documents retrieved from the corpus using the expanded query. Like the LTR method, KDE_E is based only on the re-ranking of the documents retrieved by an initial retrieval method (i.e., LM.Dir).

It is, therefore, very interesting that the KDE_E is not only very competitive against LTR and the KDE-based methods, but also with KDE+KDE_E+RMT_E. In the TREC 2013 queries, KDE_E even outperformed KDE+KDE_E+RMT_E for both top-precision metrics, P30 and Rprec. KDE+KDE_E+RMT_E outperformed the other methods on MAP, but the difference was not statistically significant against KDE_E. In the TREC 2014 queries the RMT_E-based methods outperform KDE_E on the recall-oriented metrics, MAP and Rprec. KDE+KDE_E+RMT_E statistically significantly outperformed KDE_E in the recall-oriented metrics, MAP and Rprec, in part due to the use of the RMT_E method in KDE+KDE_E+RMT_E to obtain the candidate set of documents for re-ranking.

Table 5.4: TREC 2014 dataset results.

Method	MAP	P30	Rprec
LM.Dir	0.4316	0.6315	0.4552
Recency	0.4323	0.6382	0.4576
KDE(score)	0.4205	0.6303	0.4476
KDE(rank)	0.4399	0.6406	0.4664
LTR	0.4469	0.6721	0.4625
RM_E	0.4705	0.6394	0.4890
RMT_E	$0.4738^{\ddagger}$	0.6442	$0.4927^{\dagger}$
KDE_E	$0.4643^{\ddagger *}$	$0.6776^{\ddagger}$	$0.4869^{\dagger *}$
$KDE{+}KDE_E{+}RMT_E$	$\mathbf{0.5183}^{\ddagger *}$	$\mathbf{0.6970}^{\dagger}$	$\mathbf{0.5138}^{\ddagger *}$

Symbols † and * stand for a $p < 0.05$ statistical significant improvement over KDE(score) and LTR respectively (‡ and * for $p < 0.01$).

5.4.3 Full Model Analysis

To conclude the retrieval results analysis, we examine the overall gains offered by temporal evidence from topic-based external collections. The results of the evaluation on the two TREC test datasets are summarized in Table 5.3 and Table 5.4. We present the results for three retrieval effectiveness metrics: MAP, P30, and Rprec. We found that $KDE{+}KDE_E{+}RMT_E$ can outperform non-temporal learning to rank as well as state-of-the-art temporal ranking methods.

$KDE{+}KDE_E{+}RMT_E$ statistically significantly outperforms KDE(score) in both sets of queries. MAP improved 12.3% and 23.3% in the TREC 2013 and TREC 2014 topics respectively. Additionally, for the TREC 2014 topics the MAP result improved 17.8% over KDE(rank) and was statistically significant. Although, in terms of P30, $KDE{+}KDE_E{+}RMT_E$ did not outperform KDE(rank) for the TREC 2013 queries, it outperformed KDE(score) albeit the result was not a statistically significant. In contrast, the improvements on P30 with $KDE{+}KDE_E{+}RMT_E$ on the TREC 2014 topics reached a statistically significant result of 10.6% over KDE(score) and 8.8% over KDE(rank), respectively.

$KDE{+}KDE_E{+}RMT_E$ outperforms the LTR baseline consistently across all metrics on both sets of queries. The improvements of $KDE{+}KDE_E{+}RMT_E$ in

MAP and Rprec over LTR in the TREC 2014 topics were statistically significant, 16.0% and 10.0% for MAP and Rprec, respectively.

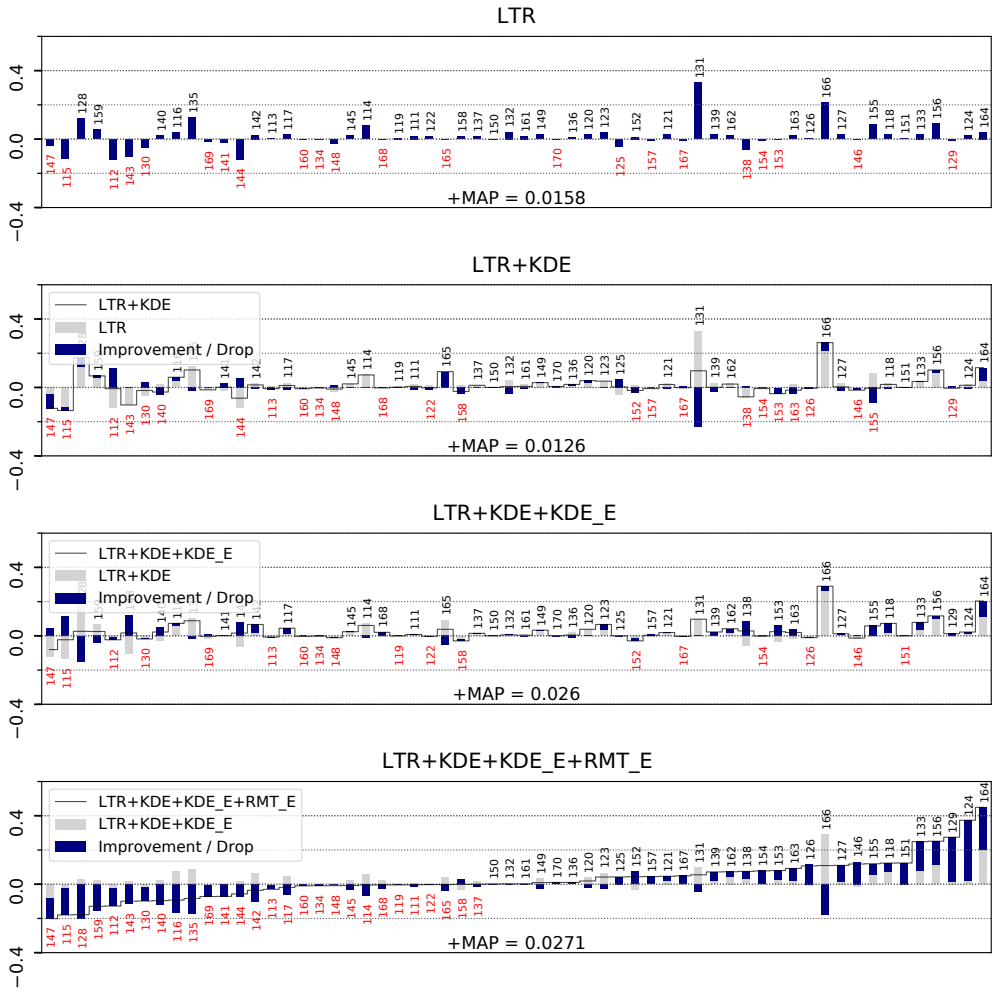


Figure 5.2: TREC 2013 – Per-feature retrieval results of the full model: graphs show Avg. Prec. relative improvements over LM.Dir by adding each temporal feature incrementally to the LM.Dir model. Each graph illustrates the per-query results, where bars are labeled with the TREC topic number. Topic labels appear above/below if the performance improved/dropped relative to LM.Dir.

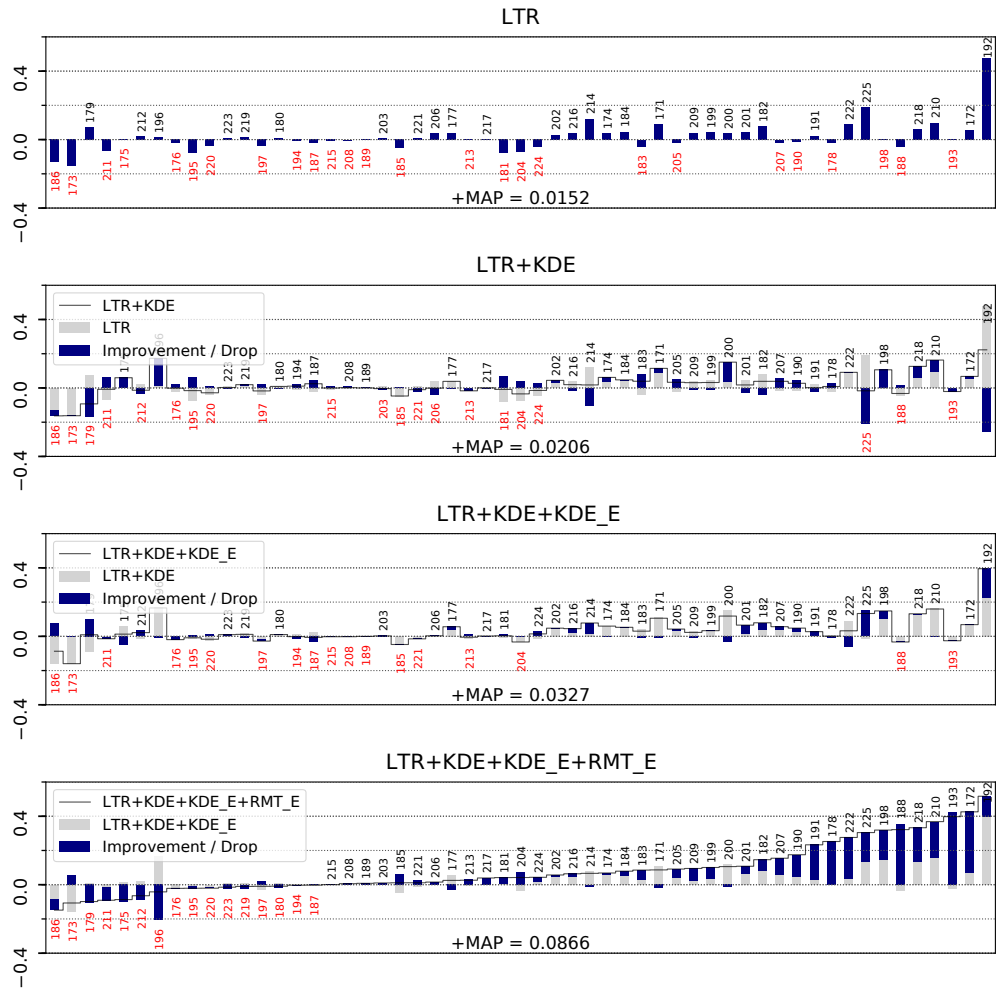


Figure 5.3: TREC 2014 – Per-feature retrieval results of the full model: graphs show Avg. Prec. relative improvements over LM.Dir by adding each temporal feature incrementally to the LM.Dir model. Each graph illustrates the per-query results, where bars are labeled with the TREC topic number. Topic labels appear above/below if the performance improved/dropped relative to LM.Dir.

5.4.4 Per-Query Analysis

In this section, we present a per-query analysis of the gains in Avg. Prec. to understand more clearly the contribution of each feature of the model. We show the progression of the results starting from the LM.Dir baseline and going up to the full model in a number of steps. The first step is the introduction of learning to rank for ranking using non-temporal features. Then, as a second step, we introduce the KDE(rank) temporal feedback feature. The third step corresponds to RMTS, which performs efficient temporal feedback on the partitioned external expansion collection. Finally, the last step corresponds to the full model. The full model uses an efficient federated pseudo-relevance feedback architecture to collect evidence for both time-aware query expansion and temporal feedback. The full model will be pictured after KDE_E so that it can be easily compared. Keep in mind that it performs the query expansion step before learning to rank.

In Figure 5.2 we present the per-query gains in Avg. Prec. relative to the LM.Dir baseline to analyze the effectiveness across the whole set of test queries for TREC 2013. As it can be seen in the LTR graph, effectiveness improves for several queries when applying learning to rank, specially on topic 131 and 166. On average, the results were not as good when introducing temporal feedback (LTR + KDE(rank)). However, we can see that one single query (topic 131) is to blame for most of it. This problem with topic 131 goes away when using RMTS because an extra source of temporal feedback is added. In fact, it can be seen that RMTS reduced the number of topics that were hurt from 27 to only 18 and improved Avg. Prec. on more topics. The full model outperformed LM.Dir on about half of the topics in the TREC 2013 dataset, and although the effectiveness was hurt for one half, the gains in performance on the other half were higher and outweighed them. Approximately one-quarter of the topics observed a substantial drop in the retrieval effectiveness as measured using MAP. Interestingly, we see that two topics only improved when using the full model. Those are topic 124 and topic 129.

In Figure 5.3 we present the per-query gains in Avg. Prec. relative to the LM.Dir baseline to analyze the effectiveness across the whole set of test queries

for TREC 2014. When using learning to rank we see that several topics improved, but what stands out in the LTR graph is that topic 192 is an outlier. When combining temporal feedback with LTR, results improve slightly, however topic 192 is hurt. RMTS was again more robust and improved results on several queries, bringing topic 192 back to the level of performance of LTR. As seen in the results for the TREC 2013 topics, RMTS also reduces the number of TREC 2014 topics that are hurt by temporal re-ranking (from 21 to 18). This number is further reduced (to 16) with the full model, which outperformed LM.Dir on about three-quarters of the test topics. Retrieval effectiveness was visibly degraded just for topic 196 compared to LM.Dir because up until this method the effectiveness was improving. We can clearly see that some of these queries under-performed when using other methods as well, since 10 out of the 16 failing queries failed with LTR alone. Again, we can see that the gains in Avg. Prec. are of a larger magnitude than the other methods. On the TREC 2014 queries, the full model was much more stable than on the TREC 2013 queries.

5.4.5 Temporal Distribution Analysis

This section aims to provide extra insights to understand the different performance of the retrieval methods in light of the effect on the temporal distribution of their top ranked documents. With this objective in mind, we look into a temporal representation of the *R-Precision* metric, Figure 5.4 and Figure 5.5: we plot the ground-truth distribution of the R relevant documents of each query (empty bars) against the relevant documents retrieved at rank depth R (shaded).

A perfect method retrieves only relevant documents, hence completely filling the empty bars. This visualization allows us to see if the methods are returning documents from the time periods that contain more relevant documents in the ground-truth. The plotted methods include the LM.Dir (no temporal evidence), KDE(rank) (temporal evidence from the corpus), KDE_E (temporal evidence from external collections), and KDE+KDE_E+RMT_E (external, temporal feedback and time-based relevance model. Additionally, we present the EMD metric to quantify the difference between the temporal distribution of the retrieved

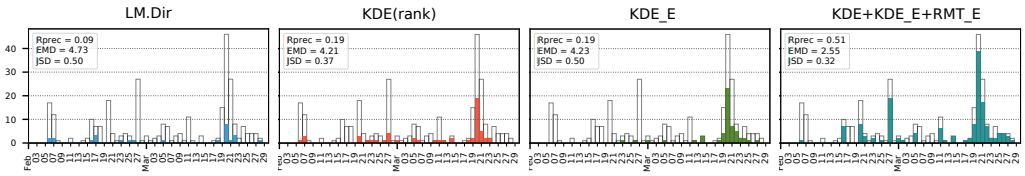
documents and the true distribution. It is interesting to observe the direct relation between the EMD and R-Precision results.

In Figure 5.4 and Figure 5.5, we plot some of the topics that have improved the most in TREC 2013 and TREC 2014, respectively. Starting with the TREC 2013 queries, we can see that for all the queries shown the temporal distribution of the top documents generally agrees with the temporal distribution of the documents in the ground truth. For the top performing topic (see Figure 5.4a) we can see that KDE_E is nudged towards retrieving documents from the most relevant time period.

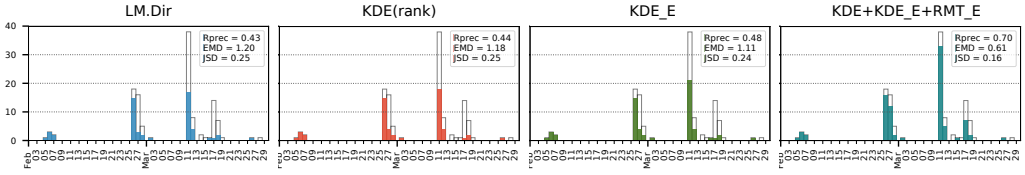
However, with RMT_E and $KDE{+}KDE_E{+}RMT_E$, which use query expansion, a second relevant time period is found. We can see that $KDE{+}KDE_E{+}RMT_E$ seems to retrieve more documents from the most relevant time period, but it retrieves some documents from this second time period as well.

In the case of topic 133 "cruise ship safety", Figure 5.4d, it is clearly visible that $KDE{+}KDE_E{+}RMT_E$ is able to focus its retrieval towards documents published in and around February 11. Inspecting the documents retrieved we found that they talked about the incident with the Carnival Triumph cruise ship. This cruise ship set sail on February 7 and three days later (February 10) it suffered a fire in the engine room.

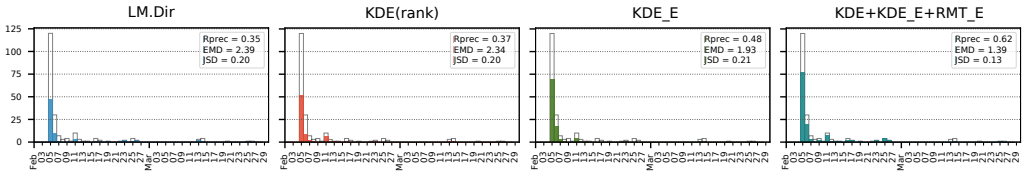
The temporal distribution of the ground truth for topic 178 "Tiger Woods regains title", Figure 5.5c, indicates that most of the relevant documents are near the time of the query. The LM.Dir and RMT_E methods retrieve documents with a similar temporal distribution to the ground truth. Nevertheless, the temporal distribution of the documents retrieved with KDE_E and $KDE{+}KDE_E{+}RMT_E$ shows that it retrieves even more documents from the most relevant day.



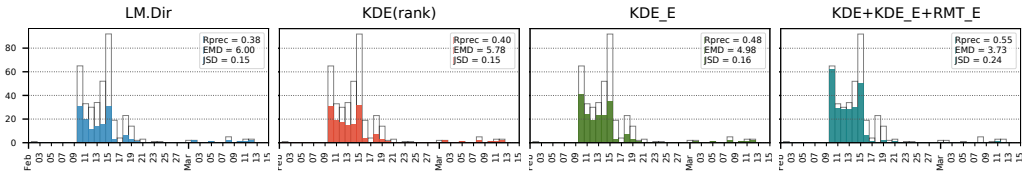
(a) MB124 – "celebrity DUI"



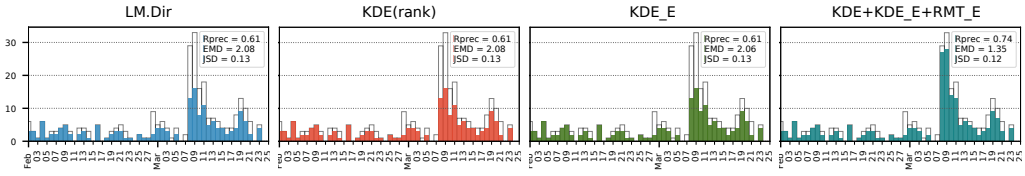
(b) MB129 – "Angry Birds cartoon"



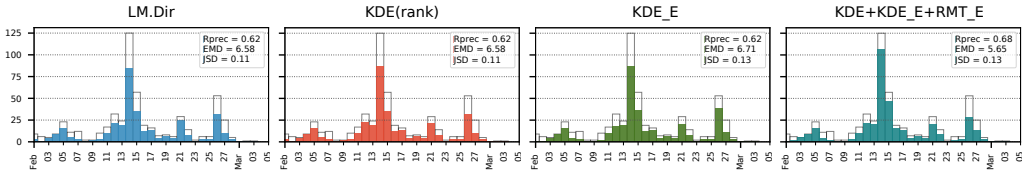
(c) MB164 – "Lindsey Vonn sidelined"



(d) MB133 – "cruise ship safety"

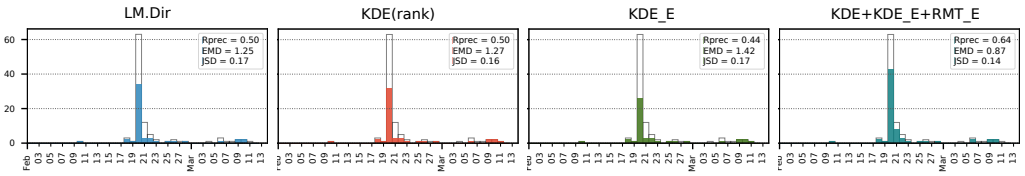


(e) MB146 – "GMO labeling"

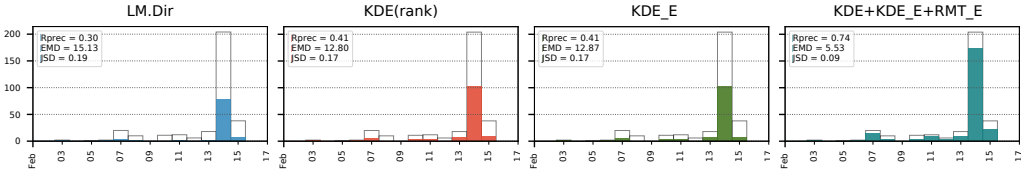


(f) MB127 – "Hagel nomination filibustered"

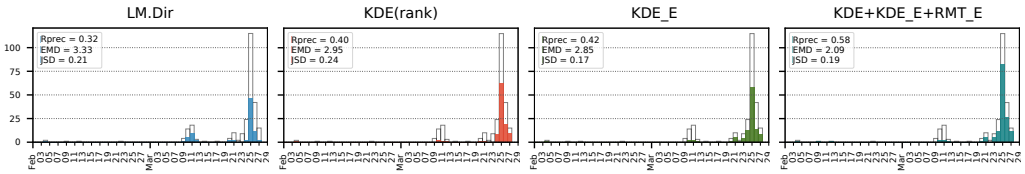
Figure 5.4: TREC 2013 – Temporal profiles of queries and and Rprec. The portion of relevant documents retrieved at a depth of R is filled.



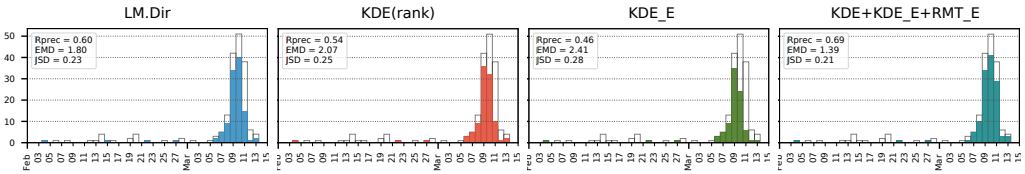
(a) MB193 – "Bulgarian protesters self immolate"



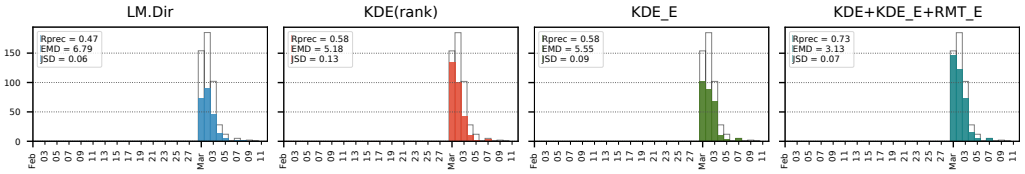
(b) MB172 – "Merging of US Air and American"



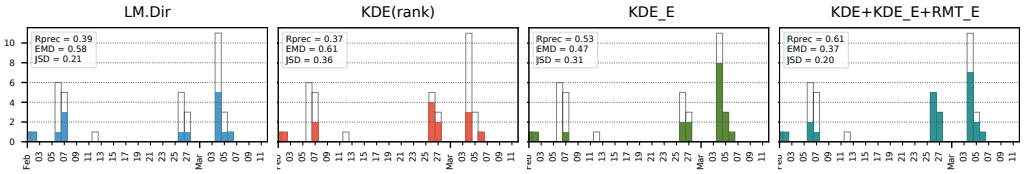
(c) MB178 – "Tiger Woods regains title"



(d) MB222 – "Oz, The Great and Powerful opens"



(e) MB218 – "Sinkhole rescues"



(f) MB225 – "Barbara Walters, chicken pox"

Figure 5.5: TREC 2014 – Temporal profiles of queries and Rprec. The portion of relevant documents retrieved at a depth of R is filled.

5.5 Summary

This chapter presented a time-aware and topic-aware pseudo-relevance feedback framework that mines textual and temporal signals from multiple information sources on Twitter. It explores the signals from verified accounts posts on Twitter, and temporal feedback to estimate the temporal relevance of search topics. The information streams from the verified accounts are automatically partitioned into verticals according to their topic. Using both query expansion and time-aware ranking this model integrates lexical, Twitter domain and temporal evidence from multiple sources of information.

Time-aware topical-based evidence mining. The results of the experiments confirmed our hypothesis that jointly modeling the topicality and temporality improves the estimation of relevance models, and yields improvements in Rprec over the timeline.

Efficient use of external collections. Building on recent advances, we show how to exploit the temporal heterogeneity of multiple external information verticals for time-aware ranking. These topic-based external verticals are exploited at two stages of the retrieval process: query expansion term selection, and temporal feedback for results re-ranking.

In the concluding chapter that follows, we explain how our approach may be generalized as a solution to some difficult information retrieval systems. Moreover, we also list directions that could be explored in future research.

CONCLUSIONS

6

"Scientists may blaze the path;
But engineers will pave it."

— Author Unknown

In this dissertation we presented strategies for improving microblog search that leverage temporal relevance estimation, multiple external Web sources of lexical and temporal evidences, and the federated search architecture. The approaches followed distinguish themselves from preceding methods mainly by dropping offline preprocessing steps while still improving efficiency in comparison to similar approaches. This approach is more in tune with the "in the moment" nature of social media, where there is an enormous variety of ephemeral conversation topics. Moreover, it also contributes to attenuate several well-known limitations of current retrieval models that for long have been identified in research, such as the high mismatch between the terms users employ to specify their information needs and the terms in relevant documents, and response-time requirements.

6.1 Thesis Summary

The major contribution of this thesis, presented in Chapter 5, is a generalized formal model for integrating both lexical and temporal evidences from external collections to improve microblog search relevance ranking. We take a federated information retrieval approach to tackle two major microblog search challenges: query expansion and time-aware ranking. We arrive at this model by formalizing the findings and expanding the work on these two chapters: In Chapter 3, we discuss a novel query expansion method over a distributed query expansion index for feedback using a federated search architecture and analyze its impact on both efficiency and effectiveness; and in Chapter 4 we discussed the impact

of using multiple external information sources to estimate the relevant time periods for a query and the integration of multiple sources of temporal evidence for time-aware ranking.

6.1.1 Federated Query Expansion

A significant finding in this thesis is the importance of reducing the computational cost of query expansion while maintaining the same retrieval precision as standard pseudo-relevance feedback. Query expansion can be essential to obtain good search results in microblog search by increasing recall, retrieval effectiveness, and better document ranking, which is essential to cope with the short queries and posts. However, to compute the expansion terms for a query using PRF, it is necessary to issue an additional, more computationally complex, initial retrieval over the whole collection, which will negatively affect response-time.Caching search results and posting lists is the strategy followed by most production retrieval systems to alleviate efficiency concerns, but this was not an option for us since typical issues with caching in static collections are exacerbated in dynamic collections. Thus, we proposed, in Chapter 3, performing query expansion with external and dynamic vertical corpora.

Federated QE. Our research has found that generally, partitioning news sources into verticals, the federated query expansion approach, can achieve similar effectiveness to standard PRF at a fraction of the computational cost. Furthermore, we found that this approach can surpass the retrieval effectiveness of using the non-partitioned news index (PRF.news) and also the whole search index (PRF), especially when the expansion corpus is live and has new and reliable documents arriving in a streaming fashion. The results show that using Wikipedia for feedback is not as effective as the proposed approach.

Cost-effective PRF. PRVF (taily) was more robust than other approaches while using on average fewer verticals and achieved one of the best results in effectiveness metrics for both the TREC 2013 and TREC 2014 query sets. Across most metrics PRVF (taily) obtained the best balance between efficiency and effectiveness. Although PRVF(crcs3) performed better in terms of MAP on TREC

2013, this had a slightly higher computational cost. Therefore, we conclude that resource selection algorithms that dynamically limit the number of verticals searched are more suitable for this task.

Quality of the query expansion corpus. We also found that news sources have a good coverage of the users' interests and can be a good external corpus for query expansion in microblogs. News sources are highly dynamic and are always kept up-to-date with current events and their unfolding over time. Therefore, it is in tune with the information seeking behavior of users in the microblog search scenario – discussion topics on Twitter are frequently related to real-world events that burst on the news and other media (Java *et al.*, 2007), and are, in essence, headline news or persistent news (Kwak *et al.*, 2010).

6.1.2 Temporal Signals from Multiple Sources

The findings in this work support the usefulness of mining the behavioral dynamics signals of the crowds. We leveraged on the hypothesis that it would be possible to improve the estimation of temporal relevance for time-sensitive queries from the correlation that exists between real-world events and activity on the Web: when an event occurs it sparks a higher volume of tweets, new visits and edits to related Wikipedia pages, and news published.This led to the proposal of a novel time-aware ranking model, in Chapter 4, that ranks results according to a predicted temporal relevance mined from multiple sources: Wikipedia (through page views and page edit history), news articles, and Twitter feedback.

Effective temporal re-ranking. The experiments we carried confirmed that combining multiple temporal signals outperforms using the corpus or a single source. The results of this investigation show that our proposed ranking model, RMTS, statistically significantly outperformed a query-likelihood model by 13.2% and outperformed a strong learning to rank baseline with several lexical and domain features by 6.2%. Our system was evaluated using the experimental setup for microblog search evaluation campaigns of TREC 2013 and TREC 2014 and was compared to three standard methods (BM25, IDF, and LM.Dir), two temporal ranking methods (Recency and KDE), and a learning to rank method.

RMTS is more reliable. A key advantage of the proposed RMTS model is its robustness and stability. It takes advantage of the signals from Wikipedia (through page views and page edit history), news articles, and Twitter feedback to estimate the temporal relevance of search topics. Temporal crowd signals were the starting point of our research, and we ended-up confirming this hypothesis that real-world events originate a burst of simultaneous Web activity in multiple, likely heterogeneous, data sources. The improvement over the LTR model (non-temporal features) could not be pin-pointed to a single source of temporal evidence. Moreover, the retrieval model is tolerant to missing temporal sources.

Unified representation. The proposed framework offers a principled methodology for mining and representing temporal signals from multiple heterogeneous sources, such as text features, domain-specific features (e.g., number of hashtags) and temporal features extracted from crowd behaviors. It allows predicting temporal relevance from heterogeneous pairs of timestamps and weights mined from multiple sources. This approach contrasts with typical LTR models, that consider only features extracted from the target corpus itself, by combining information from external sources into a single unified time-aware retrieval model. Our results have shown that the factorization of crowd temporal information into a rich set of temporal signals, provides a more robust way to disambiguate which days are more relevant for a search query.

Wikipedia temporal signals. Previous works exploit article views statistics (Ciglan and Nørvåg, 2010) and edit history (Georgescu *et al.*, 2013; Steiner *et al.*, 2013) for detecting events and entities related to the events. In fact, using Wikipedia as an external knowledge base to improve information retrieval has for some time been a hot research topic, mainly as an alternative to other domain specific thesauri. However, to the best of our knowledge this is the first work that explores the use of Wikipedia for time-aware ranking.

Our research took advantage of the always up-to-date nature of Wikipedia and found it to be a useful source of external information that can aid microblog search engines. In our evaluation we have found several examples of this correlation between topics that burst on microblogs and the higher volume of page views and edits for related Wikipedia articles.

6.1.3 Temporal Verticals

Our studies also revealed that the accuracy improvements of query expansion and temporal re-ranking can be additive. This finding is covered in Chapter 5 and is a significant finding emerging from this research. Our proposed method, PRVTM, a time-aware pseudo-relevance feedback framework that mines textual and temporal signals from multiple information sources, efficiently integrates all the features that we have found that can improve microblog search into a ranking model based on learning to rank. Our evaluation has shown that PRVTM is competitive retrieval method that can suppress some challenges that have been found in microblog search research.

Efficient query expansion. Since news sources can be crawled in real-time from the Twitter timeline, they are an important source of information for query expansion. Partitioning the incoming stream of news into different topic-based index shards and using the proposed federated query expansion architecture can provide an efficient query expansion process.

Effective temporal ranking. The findings suggest that in general the use of a federated search architecture to select multiple verticals for temporal feedback provides efficient temporal ranking. The results show that PRVTM outperformed the other methods in retrieval precision metrics.

6.2 Significance of the Work

The field of information retrieval progresses with the continuous efforts rising from either the academy and industry as the computing ecosystem shifts. Query expansion based on the pseudo-relevance feedback framework has been broadly studied in information retrieval, but we believe this dissertation contributes to a new paradigm of search in social media that is rising. This new paradigm is anchored on the requirement of more robust query processing and the need for modeling time in the matching of documents to social media queries.

Query modeling is often used to better capture users' information needs and to shorten the gap between the query and the documents to be retrieved. It

aims to reformulate simple queries to better representations of the underlying information needs by re-weighting the terms in the original queries to emphasize important terms, finding synonyms or alternative morphological forms of terms in the queries, adding additional terms to express the search intent more accurately, among other strategies. However, it is essential to avoid adding irrelevant terms to the query, an issue especially critical when expanding with external corpora. In this case, the selection of good external expansion corpora is crucial to avoid bad expansion terms that hurt retrieval precision.

Moreover, with query expansion also comes the issue of retrieval latency. This is such a pressing issue because research suggests that even slightly higher response times can affect the users' perceptions of the systems quality as a whole (Teevan *et al.*, 2013). Therefore, research in information retrieval has also devoted significant time and effort to the task of reducing the time between the issuing of the query and the search engine's response. Over the past years, efficiency research has focused not only on query processing strategies, but also on efficient indexing and storage. Nevertheless, this search for better response times should not hurt results.

We believe that by combining multiple temporal signals and having queries segmented into different verticals, a good balance between quality of results in terms of both precision and recall was achieved. Our approach is built on top of two strands of earlier research work in the setting of social media search. The first is a line of research that proposes to obtain richer query models by looking not only to the corpus but also to external collections. The second is a line of research that recognizes the importance of external temporal information.

The collections used for query expansion in this thesis were obtained from Twitter. We explored two document allocation policies for assigning documents to different verticals: 1) Source-based, and 2) Document-based. In the case of source-based allocation, all documents posted by the same source are assigned to the same vertical. The assignments of sources to verticals was done manually, however alternative methods could be explored in future work.

Applying a document-based approach to an unlabeled target collection or a larger external corpus will require new research in short text clustering

and topic modeling. Clustering short text is a difficult problem due to the vocabulary sparsity of the documents. New approaches that go beyond the usual representation of documents are needed for effective clustering. Some kind of document expansion can be the solution for the poor performance. Document representations that take into account term co-occurrence in the collection seem to be the way forward with new research aiming for more relaxed representation of sentences. However, it looks like more and more researchers are exploiting the compositionality of word embeddings to come up with representations for sentences from the combination of word embeddings.

6.3 Directions for Future Work

The research presented in this thesis lays the groundwork for future work on the introduction of interactivity in vertical-based query expansion. This interactivity can be introduced via a user interface that presents to the user the suggested verticals for a query, according to a resource selection algorithm. This approach could be seen as a hybrid relevance feedback strategy, where instead of selecting individual documents, users are able to select the preferred vertical from a list to bias the query expansion process towards the documents on that vertical.

A production federated query expansion system. It is usually easier to develop and debug a monolithic software piece running on a single hardware machine than designing a distributed system to implement the same functionality. In this case, the choice of a distributed information retrieval architecture is justified. A simple implementation uses just two types of nodes: a *broker*, and *index nodes*. The nodes need not only to exchange messages when executing user commands and queries but also to maintain the indexes coherent during real-time indexing. Most importantly, index nodes must include a mechanism to update term statistics and propagate them to other nodes, specially to allow the *broker* to merge the results from multiple *index nodes*. I see two alternative options to implement such a system. The first option is to store global statistics at the *broker* node, which means that during the merging phase the broker has

to re-score the documents using the global statistics that only it has access to. This option is perhaps the simplest, but the *broker* can be overwhelmed because it has to score the concatenation of N ranked lists of documents. Also, work is duplicated by scoring the documents at the *index nodes* for retrieval, using the statistics local to the node to provide an approximate ranking, and then again at the *broker* to merge the results into a final global ranking.

In option 2, the indexing of new documents at any of the *index nodes* must trigger the propagation of statistics for the set of unique terms contained in the documents to all the other *index nodes*. Therefore, the number of messages exchanged in the network is probably larger than option 1. Consider for example that term statistics are stored on a distributed key–value store that involves all *index nodes* with eventual consistency. Then each node is able to calculate global scores for their matching documents. The *broker* can then use a fast merge algorithm such as the *k-way merge algorithm* to merge the lists returned by *index nodes* that are involved in the search.

It is harder to imagine how such a system could answer queries using relevance models efficiently. That is, using the collection as both expansion corpus and target collection and hitting the *index nodes* only once per query. The relevance model cannot be estimated at the *broker* only as it requires an initial ranking of documents. Therefore, the original query is issued to the *index nodes* and relevance models need to be estimated by each, which leads to a different relevance model for each node. After retrieval, these relevance models have to be returned along with the results so that the *broker* can merge the results and re-rank them using a uniform combination of all the relevance models. The documents' forward indexes then need to be retrieved as well, so that the combined relevance model can be calculated efficiently at the *broker*.

Bibliography

Alonso, O., M. Gertz, and R. Baeza-Yates. 2007. "On the Value of Temporal Information in Information Retrieval." *SIGIR Forum*. 41(2): 35–41. ISSN: 0163-5840. DOI: 10.1145/1328964.1328968.

Aly, R., D. Hiemstra, and T. Demeester. 2013. "Taily: Shard Selection Using the Tail of Score Distributions." In: *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '13*. New York, NY, USA: ACM. 673–682. ISBN: 978-1-4503-2034-4. DOI: 10.1145/2484028.2484033.

Amodeo, G., G. Amati, and G. Gambosi. 2011. "On Relevance, Time and Query Expansion." In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. *CIKM '11*. New York, NY, USA: ACM. 1973–1976. ISBN: 978-1-4503-0717-8. DOI: 10.1145/2063576.2063868.

Anderson, C. 2006. *The Long Tail: Why the Future of Business Is Selling Less of More*. en. Hachette UK. ISBN: 978-1-4013-8463-0.

Arguello, J., J. Elsas, J. Callan, and J. Carbonell. 2008. "Document Representation and Query Expansion Models for Blog Recommendation." English. In: *Proceedings of the 2nd International Conference on Weblogs and Social Media*. *ICWSM '08*. 10–18. ISBN: 978-1-57735-355-3.

Arguello, J., F. Diaz, and J. Callan. 2011. "Learning to Aggregate Vertical Results into Web Search Results." In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. *CIKM '11*. New York, NY, USA: ACM. 201–210. ISBN: 978-1-4503-0717-8. DOI: 10.1145/2063576.2063611.

Asadi, N. 2013. "Multi-Stage Search Architectures for Streaming Documents." *Doctoral dissertation*. University of Maryland (College Park, Md.)

Baeza-Yates, R. A. and B. Ribeiro-Neto. 1999. *Modern Information Retrieval*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc. ISBN: 978-0-201-39829-8.

Bendersky, M., D. Metzler, and W. B. Croft. 2012. "Effective Query Formulation with Multiple Information Sources." In: *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. *WSDM '12*. New York, NY, USA: ACM. 443–452. ISBN: 978-1-4503-0747-5. DOI: 10.1145/2124295.2124349.

Berberich, K. and S. Bedathur. 2013. "Temporal Diversification of Search Results." In: *SIGIR 2013 Workshop on Time-Aware Information Access (TAIA 2013)*.

Berberich, K., S. Bedathur, O. Alonso, and G. Weikum. 2010. "A Language Modeling Approach for Temporal Information Needs." en. In: *Advances in Information Retrieval*. *ECIR '10*. Springer, Berlin, Heidelberg. 13–25. ISBN: 978-3-642-12274-3 978-3-642-12275-0. DOI: 10.1007/978-3-642-12275-0_5.

Broder, A. 2002. "A Taxonomy of Web Search." *SIGIR Forum*. 36(2): 3–10. ISSN: 0163-5840. DOI: 10.1145/792550.792552.

Broder, A. Z., D. Carmel, M. Herscovici, A. Soffer, and J. Zien. 2003. "Efficient Query Evaluation Using a Two-Level Retrieval Process." In: *Proceedings of the Twelfth International Conference on Information and Knowledge Management*. *CIKM '03*. New York, NY, USA: ACM. 426–434. ISBN: 978-1-58113-723-1. DOI: 10.1145/956863.956944.

Buntain, C., J. Lin, and J. Golbeck. 2016. "Discovering Key Moments in Social Media Streams." In: *2016 13th IEEE Annual Consumer Communications Networking Conference (CCNC)*. 366–374. DOI: 10.1109/CCNC.2016.7444808.

Cai, F., S. Liang, and M. de Rijke. 2014. "Time-Sensitive Personalized Query Auto-Completion." In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. *CIKM '14*. New York, NY, USA: ACM. 1599–1608. ISBN: 978-1-4503-2598-1. DOI: 10.1145/2661829.2661921.

Campos, R., G. Dias, A. M. Jorge, and C. Nunes. 2017. "Identifying Top Relevant Dates for Implicit Time Sensitive Queries." en. *Information Retrieval Journal*. May: 1–36. ISSN: 1386-4564, 1573-7659. DOI: 10.1007/s10791-017-9302-1.

Carpineto, C. and G. Romano. 2012. "A Survey of Automatic Query Expansion in Information Retrieval." *ACM Comput. Surv.* 44(1): 1:1–1:50. ISSN: 0360-0300. DOI: 10.1145/2071389.2071390.

Cartright, M.-A., J. Allan, V. Lavrenko, and A. McGregor. 2010. "Fast Query Expansion Using Approximations of Relevance Models." In: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. *CIKM '10*. New York, NY, USA: ACM. 1573–1576. ISBN: 978-1-4503-0099-5. DOI: 10.1145/1871437.1871675.

Chang, Y., A. Dong, P. Kolari, R. Zhang, Y. Inagaki, F. Diaz, H. Zha, and Y. Liu. 2013. "Improving Recency Ranking Using Twitter Data." *ACM Trans. Intell. Syst. Technol.* 4(1): 4:1–4:24. ISSN: 2157-6904. DOI: 10.1145/2414425.2414429.

Chen, Q., Q. Hu, J. Huang, and L. He. 2018. "TAKer: Fine-Grained Time-Aware Microblog Search with Kernel Density Estimation." *IEEE Transactions on Knowledge and Data Engineering*. PP(99): 1–1. ISSN: 1041-4347. DOI: 10.1109/TKDE.2018.2794538.

Cheng, S., A. Arvanitis, and V. Hristidis. 2013. "How Fresh Do You Want Your Search Results?" In: *Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management*. *CIKM '13*. New York, NY, USA: ACM. 1271–1280. ISBN: 978-1-4503-2263-8. DOI: 10.1145/2505515.2505696.

Choi, J. and W. B. Croft. 2012. "Temporal Models for Microblogs." In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. *CIKM '12*. New York, NY, USA: ACM. 2491–2494. ISBN: 978-1-4503-1156-4. DOI: 10.1145/2396761.2398674.

Choi, J., W. B. Croft, and J. Y. Kim. 2012. "Quality Models for Microblog Retrieval." In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. *CIKM '12*. New York, NY, USA: ACM. 1834–1838. ISBN: 978-1-4503-1156-4. DOI: 10.1145/2396761.2398527.

Ciglan, M. and K. Nørvåg. 2010. "WikiPop: Personalized Event Detection System Based on Wikipedia Page View Statistics." In: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. *CIKM '10*. New York, NY, USA: ACM. 1931–1932. ISBN: 978-1-4503-0099-5. DOI: 10.1145/1871437.1871769.

Costa, M., F. Couto, and M. Silva. 2014. "Learning Temporal-Dependent Ranking Models." In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. *SIGIR '14*. New York, NY, USA: ACM. 757–766. ISBN: 978-1-4503-2257-7. DOI: 10.1145/2600428.2609619.

Dai, N. and B. D. Davison. 2010. "Freshness Matters: In Flowers, Food, and Web Authority." In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '10*. New York, NY, USA: ACM. 114–121. ISBN: 978-1-4503-0153-4. DOI: 10.1145/1835449.1835471.

Dai, N., M. Shokouhi, and B. D. Davison. 2011. "Learning to Rank for Freshness and Relevance." In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '11*. New York, NY, USA: ACM. 95–104. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2009933.

Dakka, W., L. Gravano, and P. Ipeirotis. 2012. "Answering General Time-Sensitive Queries." *IEEE Transactions on Knowledge and Data Engineering*. 24(2): 220–235. ISSN: 1041-4347. DOI: 10.1109/TKDE.2010.187.

Diaz, F. 2009. "Integration of News Content into Web Results." In: *Proceedings of the Second ACM International Conference on Web Search and Data Mining*. *WSDM '09*. New York, NY, USA: ACM. 182–191. ISBN: 978-1-60558-390-7. DOI: 10.1145/1498759.1498825.

Diaz, F. 2015. "Condensed List Relevance Models." In: *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*. *ICTIR '15*. New York, NY, USA: ACM. 313–316. ISBN: 978-1-4503-3833-2. DOI: 10.1145/2808194.2809491.

Diaz, F. and R. Jones. 2004. "Using Temporal Profiles of Queries for Precision Prediction." In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '04*. New York, NY, USA: ACM. 18–24. ISBN: 1-58113-881-4. DOI: 10.1145/1008992.1008998.

Diaz, F. and D. Metzler. 2006. "Improving the Estimation of Relevance Models Using Large External Corpora." In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '06*. New York, NY, USA: ACM. 154–161. ISBN: 1-59593-369-7. DOI: 10.1145/1148170.1148200.

Dong, A., Y. Chang, Z. Zheng, G. Mishne, J. Bai, R. Zhang, K. Buchner, C. Liao, and F. Diaz. 2010. "Towards Recency Ranking in Web Search." In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. *WSDM '10*. New York, NY, USA: ACM. 11–20. ISBN: 978-1-60558-889-6. DOI: 10.1145/1718487.1718490.

Efron, M. 2010. "Hashtag Retrieval in a Microblogging Environment." In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '10*. New York, NY, USA: ACM. 787–788. ISBN: 978-1-4503-0153-4. DOI: 10.1145/1835449.1835616.

Efron, M. 2011. "Information Search and Retrieval in Microblogs." en. *Journal of the American Society for Information Science and Technology*. 62(6): 996–1008. ISSN: 1532-2890. DOI: 10.1002/asi.21512.

Efron, M. and G. Golovchinsky. 2011. "Estimation Methods for Ranking Recent Information." In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '11*. New York, NY, USA: ACM. 495–504. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2009984.

Efron, M., J. Lin, J. He, and A. de Vries. 2014. "Temporal Feedback for Tweet Search with Non-Parametric Density Estimation." In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. *SIGIR '14*. New York, NY, USA: ACM. 33–42. ISBN: 978-1-4503-2257-7. DOI: 10.1145/2600428.2609575.

Elsas, J. L., J. Arguello, J. Callan, and J. G. Carbonell. 2008. "Retrieval and Feedback Models for Blog Feed Search." In: *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '08*. New York, NY, USA: ACM. 347–354. ISBN: 978-1-60558-164-4. DOI: 10.1145/1390334.1390394.

Elsas, J. L. and S. T. Dumais. 2010. "Leveraging Temporal Dynamics of Document Content in Relevance Ranking." In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. *WSDM '10*. New York, NY, USA: ACM. 1–10. ISBN: 978-1-60558-889-6. DOI: 10.1145/1718487.1718489.

Endres, D. M. and J. E. Schindelin. 2006. "A New Metric for Probability Distributions." *IEEE Trans. Inf. Theor.* 49(7): 1858–1860. ISSN: 0018-9448. DOI: 10.1109/TIT.2003.813506.

Fan, F., R. Qiang, C. Lv, and J. Yang. 2015. "Improving Microblog Retrieval with Feedback Entity Model." In: *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. *CIKM '15*. New York, NY, USA: ACM. 573–582. ISBN: 978-1-4503-3794-6. DOI: 10.1145/2806416.2806461.

Ferragina, P. and U. Scaiella. 2010. "TAGME: On-the-Fly Annotation of Short Text Fragments (by Wikipedia Entities)." In: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. *CIKM '10*. New York, NY, USA: ACM. 1625–1628. ISBN: 978-1-4503-0099-5. DOI: 10.1145/1871437.1871689.

Ferron, M. and P. Massa. 2012. "Psychological Processes Underlying Wikipedia Representations of Natural and Manmade Disasters." In: *Proceedings of the Eighth Annual International Symposium on Wikis and Open Collaboration*. *WikiSym '12*. New York, NY, USA: ACM. 2:1–2:10. ISBN: 978-1-4503-1605-7. DOI: 10.1145/2462932.2462935.

Furnas, G. W., T. K. Landauer, L. M. Gomez, and S. T. Dumais. 1987. "The Vocabulary Problem in Human-System Communication." *Commun. ACM*. 30(11): 964–971. ISSN: 0001-0782. DOI: 10.1145/32206.32212.

Georgescu, M., N. Kanhabua, D. Krause, W. Nejdl, and S. Siersdorfer. 2013. "Extracting Event-Related Information from Article Updates in Wikipedia." en. In: *Advances in Information Retrieval. ECIR '13*. Springer, Berlin, Heidelberg. 254–266. ISBN: 978-3-642-36972-8 978-3-642-36973-5. DOI: 10.1007/978-3-642-36973-5_22.

Gonçalves, G., F. Martins, and J. Magalhães. 2017. "NOVASearch at TREC 2017 Real-Time Summarization Track." In: *Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC 2017, Gaithersburg, Maryland, USA, November 15-17, 2017*. Ed. by E. M. Voorhees and A. Ellis. Vol. Special Publication 500-324. National Institute of Standards and Technology (NIST).

Gonçalves, G., F. Martins, and J. Magalhães. 2018. "Analysis of Subtopic Discovery Algorithms for Real-Time Information Summarization." In: *Companion Proceedings of the The Web Conference 2018. WWW '18*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 1855–1856. ISBN: 978-1-4503-5640-4. DOI: 10.1145/3184558.3191651.

Guo, W., H. Li, H. Ji, and M. Diab. 2013. "Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media." In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics. 239–249.

Gupta, D. and K. Berberich. 2014. "Identifying Time Intervals of Interest to Queries." In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. CIKM '14*. New York, NY, USA: ACM. 1835–1838. ISBN: 978-1-4503-2598-1. DOI: 10.1145/2661829.2661927.

Gupta, D. and K. Berberich. 2015. "Temporal Query Classification at Different Granularities." In: *Proceedings of the 22Nd International Symposium on String Processing and Information Retrieval - Volume 9309. SPIRE 2015*. New York, NY, USA: Springer-Verlag New York, Inc. 156–164. ISBN: 978-3-319-23825-8. DOI: 10.1007/978-3-319-23826-5_16.

Hafizoglu, F., E. C. Kucukoglu, and I. S. Altingovde. 2017. "On the Efficiency of Selective Search." en. In: *Advances in Information Retrieval*. *ECIR '17*. Springer, Cham. 705–712. DOI: 10.1007/978-3-319-56608-5_69.

Hanbury, A., H. Müller, K. Balog, T. Brodt, G. V. Cormack, I. Eggel, T. Gollub, F. Hopfgartner, J. Kalpathy-Cramer, N. Kando, A. Krithara, J. Lin, S. Mercer, and M. Potthast. 2015. "Evaluation-as-a-Service: Overview and Outlook." *arXiv:1512.07454 [cs]*. Dec. arXiv: 1512.07454 [cs].

Hofmann, T. 1999. "Probabilistic Latent Semantic Analysis." In: *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*. *UAI'99*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 289–296. ISBN: 1-55860-614-9.

Hopfgartner, F., A. Hanbury, H. Müller, N. Kando, S. Mercer, J. Kalpathy-Cramer, M. Potthast, T. Gollub, A. Krithara, J. Lin, K. Balog, and I. Eggel. 2015. "Report on the Evaluation-as-a-Service (EaaS) Expert Workshop." *SIGIR Forum*. 49(1): 57–65. ISSN: 0163-5840. DOI: 10.1145/2795403.2795416.

Jatowt, A., C.-M. Au Yeung, and K. Tanaka. 2013. "Estimating Document Focus Time." In: *Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management*. *CIKM '13*. New York, NY, USA: ACM. 2273–2278. ISBN: 978-1-4503-2263-8. DOI: 10.1145/2505515.2505655.

Jatowt, A., Y. Kawai, and K. Tanaka. 2005. "Temporal Ranking of Search Engine Results." In: *Proceedings of the 6th International Conference on Web Information Systems Engineering*. *WISE'05*. Berlin, Heidelberg: Springer-Verlag. 43–52. ISBN: 978-3-540-30017-5. DOI: 10.1007/11581062_4.

Java, A., X. Song, T. Finin, and B. Tseng. 2007. "Why We Twitter: Understanding Microblogging Usage and Communities." In: *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*. *WebKDD/SNA-KDD '07*. New York, NY, USA: ACM. 56–65. ISBN: 978-1-59593-848-0. DOI: 10.1145/1348549.1348556.

Joho, H., A. Jatowt, and R. Blanco. 2014. "NTCIR Temporalia: A Test Collection for Temporal Information Access Research." In: *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*. *WWW Companion '14*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 845–850. ISBN: 978-1-4503-2745-9. DOI: 10.1145/2567948.2579044.

Jones, R. and F. Diaz. 2007. "Temporal Profiles of Queries." *ACM Trans. Inf. Syst.* 25(3). ISSN: 1046-8188. DOI: 10.1145/1247715.1247720.

Kanhabua, N., R. Blanco, and K. Nørvåg. 2015a. "Temporal Information Retrieval." *Foundations and Trends® in Information Retrieval*. 9(2): 91–208. ISSN: 1554-0669. DOI: 10.1561/1500000043.

Kanhabua, N., T. Ngoc Nguyen, and W. Nejdl. 2015b. "Learning to Detect Event-Related Queries for Web Search." In: *Proceedings of the 24th International Conference on World Wide Web Companion*. *WWW '15 Companion*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 1339–1344. ISBN: 978-1-4503-3473-0. DOI: 10.1145/2740908.2741698.

Kanhabua, N. and K. Nørvåg. 2010. "Determining Time of Queries for Re-Ranking Search Results." In: *Proceedings of the 14th European Conference on Research and Advanced Technology for Digital Libraries*. *ECDL'10*. Berlin, Heidelberg: Springer-Verlag. 261–272. ISBN: 978-3-642-15463-8.

Kanhabua, N. and K. Nørvåg. 2012. "Learning to Rank Search Results for Time-Sensitive Queries." In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. *CIKM '12*. New York, NY, USA: ACM. 2463–2466. ISBN: 978-1-4503-1156-4. DOI: 10.1145/2396761.2398667.

Kanhabua, N., H. Ren, and T. B. Moeslund. 2016. "Learning Dynamic Classes of Events Using Stacked Multilayer Perceptron Networks." *arXiv:1606.07219 [cs]*. June. arXiv: 1606.07219 [cs].

Kanoulas, E., K. Dai, V. Pavlu, and J. A. Aslam. 2010. "Score Distribution Models: Assumptions, Intuition, and Robustness to Score Manipulation." In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '10*. New York, NY, USA: ACM. 242–249. ISBN: 978-1-4503-0153-4. DOI: 10.1145/1835449.1835491.

Karmaker Santu, S. K., L. Li, D. H. Park, Y. Chang, and C. Zhai. 2017. "Modeling the Influence of Popular Trending Events on User Search Behavior." In: *Proceedings of the 26th International Conference on World Wide Web Companion*. *WWW '17 Companion*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 535–544. ISBN: 978-1-4503-4914-7. DOI: 10.1145/3041021.3054188.

Keikha, M., S. Gerani, and F. Crestani. 2011. "Time-Based Relevance Models." In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '11*. New York, NY, USA: ACM. 1087–1088. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2010062.

Kim, Y., J. Callan, J. S. Culpepper, and A. Moffat. 2016. "Does Selective Search Benefit from WAND Optimization?" en. In: *Advances in Information Retrieval*. *ECIR '16*. Springer, Cham. 145–158. ISBN: 978-3-319-30670-4 978-3-319-30671-1. DOI: 10.1007/978-3-319-30671-1_11.

Kim, Y., R. Yeniterzi, and J. Callan. 2012. "Overcoming Vocabulary Limitations in Twitter Microblogs." In: *Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012*. Ed. by E. M. Voorhees and L. P. Buckland. Vol. Special Publication 500-298. National Institute of Standards and Technology (NIST).

König, A. C., M. Gamon, and Q. Wu. 2009. "Click-through Prediction for News Queries." In: *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '09*. New York, NY, USA: ACM. 347–354. ISBN: 978-1-60558-483-6. DOI: 10.1145/1571941.1572002.

Kulkarni, A. and J. Callan. 2010. "Document Allocation Policies for Selective Searching of Distributed Indexes." In: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. *CIKM '10*. New York, NY, USA: ACM. 449–458. ISBN: 978-1-4503-0099-5. DOI: 10.1145/1871437.1871497.

Kulkarni, A. and J. Callan. 2015. "Selective Search: Efficient and Effective Search of Large Textual Collections." *ACM Trans. Inf. Syst.* 33(4): 17:1–17:33. ISSN: 1046-8188. DOI: 10.1145/2738035.

Kulkarni, A., J. Teevan, K. M. Svore, and S. T. Dumais. 2011. "Understanding Temporal Query Dynamics." In: *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*. *WSDM '11*. New York, NY, USA: ACM. 167–176. ISBN: 978-1-4503-0493-1. DOI: 10.1145/1935826.1935862.

Kulkarni, A., A. S. Tigelaar, D. Hiemstra, and J. Callan. 2012. "Shard Ranking and Cutoff Estimation for Topically Partitioned Collections." In: *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. *CIKM '12*. New York, NY, USA: ACM. 555–564. ISBN: 978-1-4503-1156-4. DOI: 10.1145/2396761.2396833.

Kwak, H., C. Lee, H. Park, and S. Moon. 2010. "What Is Twitter, a Social Network or a News Media?" In: *Proceedings of the 19th International Conference on World Wide Web*. *WWW '10*. New York, NY, USA: ACM. 591–600. ISBN: 978-1-60558-799-8. DOI: 10.1145/1772690.1772751.

Larkey, L. S., M. E. Connell, and J. Callan. 2000. "Collection Selection and Results Merging with Topically Organized U.S. Patents and TREC Data." In: *Proceedings of the Ninth International Conference on Information and Knowledge Management*. *CIKM '00*. New York, NY, USA: ACM. 282–289. ISBN: 1-58113-320-0. DOI: 10.1145/354756.354830.

Lavrenko, V. and J. Allan. 2006. "Real-Time Query Expansion in Relevance Models." English. *IR* No. 473. University of Massachusetts Amherst.

Lavrenko, V. and W. B. Croft. 2001. "Relevance Based Language Models." In: *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '01*. New York, NY, USA: ACM. 120–127. ISBN: 1-58113-331-6. DOI: 10.1145/383952.383972.

Li, X. and W. B. Croft. 2003. "Time-Based Language Models." In: *Proceedings of the Twelfth International Conference on Information and Knowledge Management*. *CIKM '03*. New York, NY, USA: ACM. 469–475. ISBN: 1-58113-723-0. DOI: 10.1145/956863.956951.

Liang, S., Z. Ren, W. Weerkamp, E. Meij, and M. de Rijke. 2014. "Time-Aware Rank Aggregation for Microblog Search." In: *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. *CIKM '14*. New York, NY, USA: ACM. 989–998. ISBN: 978-1-4503-2598-1. DOI: 10.1145/2661829.2661905.

Lin, J. and M. Efron. 2013a. "Evaluation As a Service for Information Retrieval." *SIGIR Forum*. 47(2): 8–14. ISSN: 0163-5840. DOI: 10.1145/2568388.2568390.

Lin, J. and M. Efron. 2013b. "Overview of the TREC-2013 Microblog Track." In: *Proceedings of The Twenty-Second Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*. Ed. by E. M. Voorhees. Vol. Special Publication 500-302. National Institute of Standards and Technology (NIST).

Lin, J. and M. Efron. 2014. "Infrastructure Support for Evaluation As a Service." In: *Proceedings of the 23rd International Conference on World Wide Web*. *WWW '14 Companion*. New York, NY, USA: ACM. 79–82. ISBN: 978-1-4503-2745-9. DOI: 10.1145/2567948.2577014.

Lin, J., M. Efron, Y. Wang, and G. Sherman. 2014. "Overview of the TREC-2014 Microblog Track." In: *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*. Ed. by E. M. Voorhees and A. Ellis. Vol. Special Publication 500-308. National Institute of Standards and Technology (NIST).

Lloyd, S. 1982. "Least Squares Quantization in PCM." *IEEE Transactions on Information Theory*. 28(2): 129–137. ISSN: 0018-9448. DOI: 10.1109/TIT.1982.1056489.

Macdonald, C., N. Tonellotto, and I. Ounis. 2012. "Learning to Predict Response Times for Online Query Scheduling." In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '12*. New York, NY, USA: ACM. 621–630. ISBN: 978-1-4503-1472-5. DOI: 10.1145/2348283.2348367.

Martins, F. and J. Magalhães. 2014. "NovaSearch at TREC 2014 Microblog Track." In: *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*.

Martins, F., J. Magalhães, and J. Callan. 2016a. "Barbara Made the News: Mining the Behavior of Crowds for Time-Aware Learning to Rank." In: *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*. *WSDM '16*. San Francisco, CA, USA: ACM.

Martins, F., J. Magalhães, and J. Callan. 2016b. "Jitter Search: A News-Based Real-Time Twitter Search Interface." en. In: *Advances in Information Retrieval*. *ECIR '16*. Springer, Cham. 841–844. ISBN: 978-3-319-30670-4 978-3-319-30671-1. DOI: 10.1007/978-3-319-30671-1_77.

Martins, F., J. Magalhães, and J. Callan. 2018. "A Vertical PRF Architecture for Microblog Search." In: *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*. *ICTIR '18*. New York, NY, USA: ACM.

Martins, F., J. Magalhães, and J. Callan. 2019. "Modeling Temporal Evidence from External Collections." In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. *WSDM '19*. Melbourne, Australia: ACM.

Martins, F., A. Mourão, and J. Magalhães. 2013. "NovaSearch at TREC 2013 Microblog Track: Experiments with Reranking Using Wikipedia." In: *Proceedings of The Twenty-Second Text REtrieval Conference, TREC 2013, Gaithersburg, Maryland, USA, November 19-22, 2013*.

Massoudi, K., M. Tsagkias, M. de Rijke, and W. Weerkamp. 2011. "Incorporating Query Expansion and Quality Indicators in Searching Microblog Posts." en. In: *Advances in Information Retrieval*. *ECIR '11*. Springer, Berlin, Heidelberg. 362–367. ISBN: 978-3-642-20160-8 978-3-642-20161-5. DOI: 10.1007/978-3-642-20161-5_36.

McCreadie, R. and C. Macdonald. 2013. "Relevance in Microblogs: Enhancing Tweet Retrieval Using Hyperlinked Documents." In: *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval*. *OAIR '13*. Paris, France, France: Le Centre de Hautes Etudes Internationales d'Informatique Documentaire. 189–196.

Metzler, D., C. Cai, and E. Hovy. 2012. "Structured Event Retrieval over Microblog Archives." In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. *NAACL HLT '12*. Stroudsburg, PA, USA: Association for Computational Linguistics. 646–655. ISBN: 978-1-937284-20-6.

Metzler, D. and W. B. Croft. 2007. "Linear Feature-Based Models for Information Retrieval." en. *Information Retrieval*. 10(3): 257–274. ISSN: 1386-4564, 1573-7659. DOI: 10.1007/s10791-006-9019-z.

Metzler, D., R. Jones, F. Peng, and R. Zhang. 2009. "Improving Search Relevance for Implicitly Temporal Queries." In: *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '09*. New York, NY, USA: ACM. 700–701. ISBN: 978-1-60558-483-6. DOI: 10.1145/1571941.1572085.

Metzler, D., T. Strohman, Y. Zhou, and W. B. Croft. 2005. "Indri at TREC 2005: Terabyte Track." In: *Proceedings of the Fourteenth Text REtrieval Conference, TREC 2005, Gaithersburg, Maryland, USA, November 15-18, 2005*. Ed. by E. M. Voorhees and L. P. Buckland. Vol. Special Publication 500-266. National Institute of Standards and Technology (NIST).

Miyanishi, T., K. Seki, and K. Uehara. 2013. "Improving Pseudo-Relevance Feedback via Tweet Selection." In: *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management*. *CIKM '13*. New York, NY, USA: ACM. 439–448. ISBN: 978-1-4503-2263-8. DOI: 10.1145/2505515.2505701.

Moffat, A., W. Webber, J. Zobel, and R. Baeza-Yates. 2007. "A Pipelined Architecture for Distributed Text Query Evaluation." en. *Information Retrieval*. 10(3): 205–231. ISSN: 1386-4564, 1573-7659. DOI: 10.1007/s10791-006-9014-4.

Naveed, N., T. Gottron, J. Kunegis, and A. C. Alhadi. 2011. "Searching Microblogs: Coping with Sparsity and Document Quality." In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. *CIKM '11*. New York, NY, USA: ACM. 183–188. ISBN: 978-1-4503-0717-8. DOI: 10.1145/2063576.2063607.

Nguyen, T. N. and N. Kanhabua. 2014. "Leveraging Dynamic Query Subtopics for Time-Aware Search Result Diversification." en. In: *Advances in Information Retrieval*. *ECIR '14*. Springer, Cham. 222–234. ISBN: 978-3-319-06027-9 978-3-319-06028-6. DOI: 10.1007/978-3-319-06028-6_19.

Nunes, S., C. Ribeiro, and G. David. 2008. "Use of Temporal Expressions in Web Search." en. In: *Advances in Information Retrieval*. *ECIR '08*. Springer, Berlin, Heidelberg. 580–584. ISBN: 978-3-540-78645-0 978-3-540-78646-7. DOI: 10.1007/978-3-540-78646-7_59.

O'Connor, B., M. Krieger, and D. Ahn. 2010. "TweetMotif: Exploratory Search and Topic Summarization for Twitter." en. In: *Fourth International AAAI Conference on Weblogs and Social Media*. 384–385.

Ogilvie, P. and J. P. Callan. 2001a. "Experiments Using the Lemur Toolkit." In: *Proceedings of The Tenth Text REtrieval Conference, TREC 2001, Gaithersburg, Maryland, USA, November 13-16, 2001*. Ed. by E. M. Voorhees and D. K. Harman. Vol. Special Publication 500-250. National Institute of Standards and Technology (NIST). 103–108.

Ogilvie, P. and J. Callan. 2001b. "The Effectiveness of Query Expansion for Distributed Information Retrieval." In: *Proceedings of the Tenth International Conference on Information and Knowledge Management*. *CIKM '01*. New York, NY, USA: ACM. 183–190. ISBN: 978-1-58113-436-0. DOI: 10.1145/502585.502617.

Page, L., S. Brin, R. Motwani, and T. Winograd. 1999. "The PageRank Citation Ranking: Bringing Order to the Web." http://ilpubs.stanford.edu:8090/422/. *Techreport*.

Peetz, M.-H., E. Meij, and M. de Rijke. 2013. "Using Temporal Bursts for Query Modeling." en. *Information Retrieval*. July: 1–35. ISSN: 1386-4564, 1573-7659. DOI: 10.1007/s10791-013-9227-2.

Peetz, M.-H. and M. de Rijke. 2013. "Cognitive Temporal Document Priors." en. In: *Advances in Information Retrieval*. *ECIR '13*. Springer, Berlin, Heidelberg. 318–330. ISBN: 978-3-642-36972-8 978-3-642-36973-5. DOI: 10.1007/978-3-642-36973-5_27.

Petri, M., J. S. Culpepper, and A. Moffat. 2013. "Exploring the Magic of WAND." In: *Proceedings of the 18th Australasian Document Computing Symposium*. *ADCS '13*. New York, NY, USA: ACM. 58–65. ISBN: 978-1-4503-2524-0. DOI: 10.1145/2537734.2537744.

Ponte, J. M. and W. B. Croft. 1998. "A Language Modeling Approach to Information Retrieval." In: *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '98*. New York, NY, USA: ACM. 275–281. ISBN: 1-58113-015-5. DOI: 10.1145/290941.291008.

Qiang, R., F. Fan, C. Lv, and J. Yang. 2015. "Knowledge-Based Query Expansion in Real-Time Microblog Search." *arXiv:1503.03961 [cs]*. Mar. arXiv: 1503.03961 [cs].

Radinsky, K., K. Svore, S. Dumais, J. Teevan, A. Bocharov, and E. Horvitz. 2012. "Modeling and Predicting Behavioral Dynamics on the Web." In: *Proceedings of the 21st International Conference on World Wide Web*. *WWW '12*. New York, NY, USA: ACM. 599–608. ISBN: 978-1-4503-1229-5. DOI: 10.1145/2187836.2187918.

Rao, J., H. He, H. Zhang, F. Ture, R. Sequiera, S. Mohammed, and J. Lin. 2017a. "Integrating Lexical and Temporal Signals in Neural Ranking Models for Searching Social Media Streams." *arXiv:1707.07792 [cs]*. July. arXiv: 1707.07792 [cs].

Rao, J. and J. Lin. 2016. "Temporal Query Expansion Using a Continuous Hidden Markov Model." In: *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*. *ICTIR '16*. New York, NY, USA: ACM. 295–298. ISBN: 978-1-4503-4497-5. DOI: 10.1145/2970398.2970424.

Rao, J., J. Lin, and M. Efron. 2015. "Reproducible Experiments on Lexical and Temporal Feedback for Tweet Search." en. In: *Advances in Information Retrieval*. *ECIR '15*. Springer, Cham. 755–767. ISBN: 978-3-319-16353-6 978-3-319-16354-3. DOI: 10.1007/978-3-319-16354-3_82.

Rao, J., X. Niu, and J. Lin. 2016. "Compressing and Decoding Term Statistics Time Series." en. In: *Advances in Information Retrieval*. *ECIR '16*. Springer, Cham. 675–681. ISBN: 978-3-319-30670-4 978-3-319-30671-1. DOI: 10.1007/978-3-319-30671-1_52.

Rao, J., F. Ture, X. Niu, and J. Lin. 2017b. "Mining the Temporal Statistics of Query Terms for Searching Social Media Posts." In: *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*. *ICTIR '17*. New York, NY, USA: ACM. 133–140. ISBN: 978-1-4503-4490-6. DOI: 10.1145/3121050.3121052.

Robertson, S. E., S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. 1994. "Okapi at TREC-3." In: *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*.

Rosa, K. D., R. Shah, B. Lin, A. Gershman, and R. Frederking. 2011. "Topical Clustering of Tweets." In: *Proceedings of the 3rd ACM Workshop on Social Web Search and Mining*. *SWSM '10*. Beijing, China: ACM.

Sakai, T. 2014. "Statistical Reform in Information Retrieval?" *SIGIR Forum*. 48(1): 3–12. ISSN: 0163-5840. DOI: 10.1145/2641383.2641385.

Sankaranarayanan, J., H. Samet, B. E. Teitler, M. D. Lieberman, and J. Sperling. 2009. "TwitterStand: News in Tweets." In: *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. *GIS '09*. New York, NY, USA: ACM. 42–51. ISBN: 978-1-60558-649-6. DOI: 10.1145/1653771.1653781.

Schilder, F. and C. Habel. 2001. "From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages." In: *Proceedings of the Workshop on Temporal and Spatial Information Processing - Volume 13*. *TASIP '01*. Stroudsburg, PA, USA: Association for Computational Linguistics. 9:1–9:8. DOI: 10.3115/1118238.1118247.

Schilder, F. and C. Habel. 2003. "Temporal Information Extraction for Temporal Question Answering." In: *New Directions in Question Answering*. 35–44.

Sequiera, R. and J. Lin. 2017. "Finally, a Downloadable Test Collection of Tweets." In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '17*. New York, NY, USA: ACM. 1225–1228. ISBN: 978-1-4503-5022-8. DOI: 10.1145/3077136.3080667.

Shokouhi, M. 2007. "Central-Rank-Based Collection Selection in Uncooperative Distributed Information Retrieval." en. In: *Advances in Information Retrieval*. *ECIR '07*. Springer, Berlin, Heidelberg. 160–172. ISBN: 978-3-540-71494-1 978-3-540-71496-5. DOI: 10.1007/978-3-540-71496-5_17.

Shokouhi, M. 2011. "Detecting Seasonal Queries by Time-Series Analysis." In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '11*. New York, NY, USA: ACM. 1171–1172. ISBN: 978-1-4503-0757-4. DOI: 10.1145/2009916.2010104.

Shokouhi, M., L. Azzopardi, and P. Thomas. 2009. "Effective Query Expansion for Federated Search." In: *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '09*. New York, NY, USA: ACM. 427–434. ISBN: 978-1-60558-483-6. DOI: 10.1145/1571941.1572015.

Shokouhi, M. and K. Radinsky. 2012. "Time-Sensitive Query Auto-Completion." In: *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '12*. New York, NY, USA: ACM. 601–610. ISBN: 978-1-4503-1472-5. DOI: 10.1145/2348283.2348364.

Shokouhi, M. and L. Si. 2011. "Federated Search." *Found. Trends Inf. Retr.* 5(1): 1–102. ISSN: 1554-0669. DOI: 10.1561/1500000010.

Si, L. and J. Callan. 2003. "Relevant Document Distribution Estimation Method for Resource Selection." In: *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. *SIGIR '03*. New York, NY, USA: ACM. 298–305. ISBN: 1-58113-646-3. DOI: 10.1145/860435.860490.

Singh, J., W. Nejdl, and A. Anand. 2016. "History by Diversity: Helping Historians Search News Archives." In: *Proceedings of the 2016 ACM on Conference on Human Information Interaction and Retrieval*. *CHIIR '16*. New York, NY, USA: ACM. 183–192. ISBN: 978-1-4503-3751-9. DOI: 10.1145/2854946.2854959.

Soboroff, I., I. Ounis, C. Macdonald, and J. Lin. 2012. "Overview of the TREC-2012 Microblog Track." In: *Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012*. Ed. by E. M. Voorhees and L. P. Buckland. Vol. Special Publication 500-298. National Institute of Standards and Technology (NIST).

Steiner, T., S. van Hooland, and E. Summers. 2013. "MJ No More: Using Concurrent Wikipedia Edit Spikes with Social Network Plausibility Checks for Breaking News Detection." In: *Proceedings of the 22Nd International Conference on World Wide Web Companion*. *WWW '13 Companion*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. 791–794. ISBN: 978-1-4503-2038-2.

Styskin, A., F. Romanenko, F. Vorobyev, and P. Serdyukov. 2011. "Recency Ranking by Diversification of Result Set." In: *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. *CIKM '11*. New York, NY, USA: ACM. 1949–1952. ISBN: 978-1-4503-0717-8. DOI: 10.1145/2063576.2063862.

Teevan, J., K. Collins-Thompson, R. W. White, S. T. Dumais, and Y. Kim. 2013. "Slow Search: Information Retrieval Without Time Constraints." In: *Proceedings of the Symposium on Human-Computer Interaction and Information Retrieval*. *HCIR '13*. New York, NY, USA: ACM. 1:1–1:10. ISBN: 978-1-4503-2570-7. DOI: 10.1145/2528394.2528395.

Teevan, J., D. Ramage, and M. R. Morris. 2011. "#TwitterSearch: A Comparison of Microblog Search and Web Search." In: *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*. *WSDM '11*. New York, NY, USA: ACM. 35–44. ISBN: 978-1-4503-0493-1. DOI: 10.1145/1935826.1935842.

Tsagkias, M., M. de Rijke, and W. Weerkamp. 2011. "Linking Online News and Social Media." In: *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*. *WSDM '11*. New York, NY, USA: ACM. 565–574. ISBN: 978-1-4503-0493-1. DOI: 10.1145/1935826.1935906.

Tunkeland, D. 2009. "A Twitter Analog to PageRank."

Vlachos, M., C. Meek, Z. Vagena, and D. Gunopulos. 2004. "Identifying Similarities, Periodicities and Bursts for Online Search Queries." In: *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*. *SIGMOD '04*. New York, NY, USA: ACM. 131–142. ISBN: 1-58113-859-8. DOI: 10.1145/1007568.1007586.

Voorhees, E. M. and L. P. Buckland, eds. 2012. *Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012*. Vol. Special Publication 500-298. National Institute of Standards and Technology (NIST).

Voorhees, E. M., J. Lin, and M. Efron. 2014. "On Run Diversity in Evaluation As a Service." In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. *SIGIR '14*. New York, NY, USA: ACM. 959–962. ISBN: 978-1-4503-2257-7. DOI: 10.1145/2600428.2609484.

Walker, E. and A. S. Nowacki. 2011. "Understanding Equivalence and Noninferiority Testing." *Journal of General Internal Medicine*. 26(2): 192–196. ISSN: 0884-8734. DOI: 10.1007/s11606-010-1513-8.

Wang, X., C. Zhai, X. Hu, and R. Sproat. 2007. "Mining Correlated Bursty Topic Patterns from Coordinated Text Streams." In: *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. *KDD '07*. New York, NY, USA: ACM. 784–793. ISBN: 978-1-59593-609-7. DOI: 10.1145/1281192.1281276.

Wang, Y. and J. Lin. 2017. "Partitioning and Segment Organization Strategies for Real-Time Selective Search on Document Streams." In: *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. *WSDM '17*. New York, NY, USA: ACM. 221–230. ISBN: 978-1-4503-4675-7. DOI: 10.1145/3018661.3018727.

Weerkamp, W., K. Balog, and M. de Rijke. 2012. "Exploiting External Collections for Query Expansion." *ACM Trans. Web*. 6(4): 18:1–18:29. ISSN: 1559-1131. DOI: 10.1145/2382616.2382621.

Weerkamp, W. and M. de Rijke. 2012. "Credibility-Inspired Ranking for Blog Post Retrieval." en. *Information Retrieval*. 15(3-4): 243–277. ISSN: 1386-4564, 1573-7659. DOI: 10.1007/s10791-011-9182-8.

Weng, J., E.-P. Lim, J. Jiang, and Q. He. 2010. "TwitterRank: Finding Topic-Sensitive Influential Twitterers." In: *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. *WSDM '10*. New York, NY, USA: ACM. 261–270. ISBN: 978-1-60558-889-6. DOI: 10.1145/1718487.1718520.

Whiting, S. and J. M. Jose. 2014. "Recent and Robust Query Auto-Completion." In: *Proceedings of the 23rd International Conference on World Wide Web*. *WWW '14*. New York, NY, USA: ACM. 971–982. ISBN: 978-1-4503-2744-2. DOI: 10.1145/2566486.2568009.

Whiting, S., I. A. Klampanos, and J. M. Jose. 2012. "Temporal Pseudo-Relevance Feedback in Microblog Retrieval." en. In: *Advances in Information Retrieval*. *ECIR '12*. Springer, Berlin, Heidelberg. 522–526. ISBN: 978-3-642-28996-5 978-3-642-28997-2. DOI: 10.1007/978-3-642-28997-2_55.

Whiting, S., K. Zhou, J. Jose, and M. Lalmas. 2013. "Temporal Variance of Intents in Multi-Faceted Event-Driven Information Needs." In: *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '13*. New York, NY, USA: ACM. 989–992. ISBN: 978-1-4503-2034-4. DOI: 10.1145/2484028.2484169.

Wurzer, D., M. Osborne, and V. Lavrenko. 2016. "Randomised Relevance Model." *arXiv:1607.02641 [cs]*. July. arXiv: 1607.02641 [cs].

Xu, T., D. W. Oard, and P. McNamee. 2014. "HLTCOE at TREC 2014: Microblog and Clinical Decision Support." In: *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*. Ed. by E. M. Voorhees and A. Ellis. Vol. Special Publication 500-308. National Institute of Standards and Technology (NIST).

Xu, Y., G. J. Jones, and B. Wang. 2009. "Query Dependent Pseudo-Relevance Feedback Based on Wikipedia." In: *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. *SIGIR '09*. New York, NY, USA: ACM. 59–66. ISBN: 978-1-60558-483-6. DOI: 10.1145/1571941.1571954.

Zhai, C. and J. Lafferty. 2001. "Model-Based Feedback in the Language Modeling Approach to Information Retrieval." In: *Proceedings of the Tenth International Conference on Information and Knowledge Management*. *CIKM '01*. New York, NY, USA: ACM. 403–410. ISBN: 1-58113-436-3. DOI: 10.1145/502585.502654.

Zhai, C. and J. Lafferty. 2004. "A Study of Smoothing Methods for Language Models Applied to Information Retrieval." *ACM Trans. Inf. Syst.* 22(2): 179–214. ISSN: 1046-8188. DOI: 10.1145/984321.984322.

Zhang, R., Y. Konda, A. Dong, P. Kolari, Y. Chang, and Z. Zheng. 2010. "Learning Recurrent Event Queries for Web Search." In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. *EMNLP '10*. Stroudsburg, PA, USA: Association for Computational Linguistics. 1129–1139.

Zhou, K., S. Whiting, J. M. Jose, and M. Lalmas. 2013. "The Impact of Temporal Intent Variability on Diversity Evaluation." en. In: *Advances in Information Retrieval*. *ECIR '13*. Springer, Berlin, Heidelberg. 820–823. ISBN: 978-3-642-36972-8 978-3-642-36973-5. DOI: 10.1007/978-3-642-36973-5_93.

TOPICAL SHARD PARTITIONING

A

We previously proposed a new method for query expansion in microblogs that uses an external corpus of Twitter posts for query expansion. The corpus was organized into a set of curated thematic verticals

$$\mathscr{C} = \{c_1 c_2 \cdots c_{|\mathscr{C}|}\} \tag{A.1}$$

by assigning each news source into its most likely category manually. This approach does not scale to a large number of sources. from which a resource selection method selects the most likely set $\mathscr{C}_q$, which are then searched in parallel. The organization of the feedback corpus into several topic-based shards (verticals) has two main objectives. Firstly, to reduce *query-drift* and *temporal query-drift* by using query-specific verticals to estimate temporal relevance and relevance models. Secondly, to improve the query response times and reduce the workload of the feedback step by searching just a few shards for each query and exploiting the parallelism afforded by distributed search on multiple nodes.

Document clustering has wide applications in information retrieval. Rosa *et al.* (2011) used k-Means and the cosine distance to cluster tweets represented by *tf-idf* feature vectors. Wang and Lin (2017) uses a document embedding based similarity metric with streaming k-Means implementation. In LDA a document can be related to multiple topics, which is undesirable for partitioning and unlikely for microblog posts. Therefore, we opt to use k-Means clustering for the creation of topic-based verticals.

A.1 Partitioning Document Streams

We employ the k-Means clustering algorithm to split social media posts into topic-based streams. Although k-Means is a tried and tested algorithm to partition documents based on semantic similarity it is often not applied online. To solve

this stream clustering problem we use mini-batch k-Means and sample-based k-Means (Kulkarni and Callan, 2015).

The sample-based approach applies the k-Means algorithm to a small subset of documents in an initial learning phase. The K cluster-centroids learned by the application of the clustering algorithm in this sample are then used to partition new documents into K topical streams. Thus, the partitioning phase can be distributed across multiple nodes.

In our case, the subset of documents (S) used in the learning phase can be extracted from a social stream collected previously. Similarly, to the sample-based k-Means approach, simple random sampling can be employed to compile a smaller more manageable sample. Since a sample (S) is used instead of the entire collection, documents outside of the sample can contain terms that were not observed in S, thus are absent from the learned cluster-centroids. In this situation, assignment of documents to topics proceeds using only the seen terms. According to Kulkarni and Callan (2015) using a relatively small sample might be sufficient to provide acceptable performance in accordance with Heaps's law.

A.2 K-Means Clustering Metrics

The k-Means algorithm requires a similarity or a distance metric for document-to-centroid assignments during the expectation step. The original formulation by Lloyd (1982) used the euclidean distance.

A.2.1 Cosine Distance

When clustering text documents using the cosine distance documents are represented as a M-dimensional *tf-idf* feature vector, where M is the size of the vocabulary. It allows clustering text documents using the similarity measured by the angular distances between *tf-idf* vector representations of documents.

The cosine distance between a document d and a centroid C^i is as follows:

$$\cos\left(\vec{C^i} \parallel \vec{d}\right) = 1 - \frac{\vec{C^i} \cdot \vec{d}}{|C^i| \times |d|} \tag{A.2}$$

A.2.2 Jensen-Shannon Divergence

The Kullback-Leibler divergence measures how one probability distribution diverges from a second probability distribution as follows:

$$KL\left(P \parallel Q\right) = \sum_i P(i) \cdot \log \frac{P(i)}{Q(i)} \tag{A.3}$$

However, this information-theoretic measure is not suitable for use with k-Means because it is asymmetric, i.e., $KL\left(P \parallel Q\right) \neq KL\left(Q \parallel P\right)$.

The Jenson-Shannon divergence (Endres and Schindelin, 2006) is a symmetrized and smoothed version of the Kullback-Leibler divergence $D_{KL}(P||Q)$ that computes the similarity between a document d and a centroid C^i as

$$JS\left(C^i \parallel d\right) = \frac{1}{2} KL\left(C^i \parallel M\right) + \frac{1}{2} KL\left(d \parallel M\right), \tag{A.4}$$

where

$$M = \frac{1}{2}\left(C^i + d\right). \tag{A.5}$$

A.2.3 Negative Kullback-Leibler Divergence

Previous work proposed metrics based on the Kullback-Leibler divergence to partition a corpus of text documents into clusters of topically similar documents. The symmetrized version of the Kullback-Leibler divergence proposed by Kulkarni and Callan (2015) for selective search, is smoothed with a background model (Ogilvie and Callan, 2001a) to compensate for the unequal sizes of documents and clusters. To employ this metric in our implementation of k-Means we first

modified it into a distance metric:

$$NKL\left(C^i \,\middle\|\, d\right) = \sum_{w \in C^i \cap d} P_C^i(w) \cdot \log \frac{\lambda \cdot P_B(w)}{P_d(w)} \tag{A.6}$$

$$+ \sum_{w \in C^i \cap d} P_d(w) \cdot \log \frac{\lambda \cdot P_B(w)}{P_C^i(w)}, \tag{A.7}$$

where $P_C^i(w)$ and $P_d(w)$ are the unigram language models of the cluster-centroid C^i and the document d, respectively. $P_B(w)$ is the probability of the term w in the background model, which is the arithmetic mean of the K centroid models. λ is the smoothing parameter. It assumes tf-only feature vectors for input since $P_B(w)$ in the numerator incorporates the inverse collection frequency of the term into the metric that Zhai and Lafferty (2004) found to behave similarly to the traditional inverse document frequency (IDF) statistic.

Using the maximum-likelihood estimate (*MLE*), the cluster-centroid is,

$$P_C^i(w) = \frac{\#(w, C^i)}{\sum_{w'} \#(w', C^i)}, \tag{A.8}$$

where $\#(w, C^i)$ is the occurrence count of w in C^i.

Following Zhai and Lafferty (2004), the document language model $P_d(w)$ is estimated using MLE with Jelinek-Mercer smoothing

$$P_d(w) = (1 - \lambda) \cdot \frac{\#(w, d)}{\sum_{w'} \#(w', d)} + \lambda \cdot P_B(w). \tag{A.9}$$

A.3 Evaluating Clustering Algorithms

We adopt two standard evaluation metrics for clustering algorithms: Normalized Mutual Info (NMI) and Adjusted Rand Index (ARI). These information theoretic evaluation scores are based only on cluster assignments.

Formally, let $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$ be the set of output clusters, and $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$ the set of labeled classes of the documents.

- *Normalized Mutual Information (NMI)*. NMI is a normalized version of

Mutual Information (MI). The normalization by the denominator $[H(\Omega) + H(\mathbb{C})]/2$ penalizes large cardinalities or subdivisions into smaller clusters. NMI maximum value is 1 for an exact match between Ω and $\mathbb{C}$. Formally:

$$NMI(\Omega, \mathbb{C}) = \frac{I(\Omega; \mathbb{C})}{[H(\Omega) + H(\mathbb{C})]/2} \tag{A.10}$$

$$= \frac{\sum_{k,j} \frac{|\omega_k \cap c_j|}{N} log \frac{|\omega_k||c_j|}{N|\omega_k \cap c_j|}}{[\sum_k \frac{|\omega_k|}{N} log \frac{|\omega_k|}{N} + \sum_j \frac{|c_j|}{N} log \frac{|c_j|}{N}]/2} \tag{A.11}$$

- *Adjusted Rand Index (ARI)* corrected-for-chance version of Rand index. Considering document clustering as a series of pairwise decisions, Rand index measures the percentage of decisions that are correct. Adjusted Rand index maximum value is 1 for an exact match between Ω and $\mathbb{C}$.

$$ARI = \frac{\sum_{k,j} \binom{|\omega_k \cap c_j|}{2} - [\sum_k \binom{|\omega_k|}{2} \sum_j \binom{|c_j|}{2}]/\binom{N}{2}}{\frac{1}{2}[\sum_k \binom{|\omega_k|}{2}) + \sum_j \binom{|c_j|}{2})] - [\sum_k \binom{|\omega_k|}{2} \sum_j \binom{|c_j|}{2}]/\binom{N}{2}} \tag{A.12}$$

Table A.1: 20 Newsgroups organization and newsgroups names.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

A.3.1 Evaluation on the 20 Newsgroups Dataset

The 20 Newsgroups dataset is a collection that contains 18846 messages collected from Usenet and distributed (nearly) evenly across 20 different newsgroups. In Table A.1 we reproduce their names and original partitioning into major themes. It is a well-known dataset for experiments in text clustering and classification.

Table A.2: 20 Newsgroups dataset evaluation results.

	k-Means		Mb k-Means		Sample k-Means	
	NMI	ARI	NMI	ARI	NMI	ARI
Euclidean	0.344	0.096	0.294	0.072	0.284	0.054
Cosine	0.360	0.212	0.359	0.224	0.307	0.175
JS	0.365	0.220	0.317	0.077	0.287	0.043
NKL	0.437	0.241	0.400	0.240	0.314	0.194
glove.6B	0.275	0.100	0.311	0.163	0.287	0.145

Table A.3: Short sentences 20 Newsgroups dataset evaluation results.

	k-Means		Mb k-Means		Sample k-Means	
	NMI	ARI	NMI	ARI	NMI	ARI
Euclidean	0.018	0.002	0.018	0.003	0.029	0.002
Cosine	0.093	0.044	0.060	0.030	0.068	0.036
JS	0.041	0.006	0.032	0.003	0.024	0.001
NKL	0.154	0.064	0.146	0.068	0.130	0.058
glove.6B	0.141	0.057	0.153	0.069	0.145	0.066

A.3.1.1 Quantitative Evaluation

The results are shown on Table A.2. It is clear that clustering with the NKL metric outperforms the other metrics in terms of NMI and ARI when clustering the whole collection using standard k-Means. Only a few standard social media datasets are available and for document clustering and classification ground-truth labels need to be available.

To simulate a short text scenario, we designed an experiment using the well-known 20 Newsgroups dataset. We built a dataset of ~208k short text documents by splitting each document into sentences using a sentence tokenizer. Each sentence is labeled using the originating document's label.

The results shown on Table A.3 show that NKL outperforms the other distance metrics in this sentence clustering task. The Jenson-Shannon divergence metric

Table A.4: 20 Newsgroups: Top clusters k-Means NKL (N=20)

1	2	3	4	5	6	7
windows	window	key	israel	car	said	space
file	server	clipper	israeli	bike	armenian	orbit
dos	widget	encryption	jews	cars	armenians	moon
files	application	chip	arab	engine	turkish	nasa
program	display	keys	jewish	just	serdar	shuttle
version	motif	government	arabs	good	turkey	earth
use	use	nsa	people	like	dead	launch
ms	x11r5	algorithm	war	ride	armenia	mission
zip	problem	escrow	muslims	oil	genocide	spacecraft
using	xterm	security	religion	riding	sig	hst

loses the second best performance to the cosine distance in this short text scenario. We believe the smoothing in *JSD* loses some discriminative power that is more important in the sentence clustering scenario.

A.3.1.2 Quality of Topics

Since the Negative Kullback-Leibler divergence metric employs smoothing the cluster-centroids could contain in its top words a few words that are very common. Therefore, to provide a better representation of the structure of the data, the top words for each cluster are extracted using the *KL-divergence* (or *relative entropy*) $KL\left(C^i \,\middle\|\, P_B\right)$ between each cluster-centroid C^i and the background model P_B, which is taken from the arithmetic mean of the K centroid models.

A selection of the topics discovered using the Negative Kullback-Leibler divergence metric over the 20 Newsgroups dataset are shown on Table A.4. The topics discovered using the D_{NKL} metric over the 20 Newsgroups dataset are shown on Table A.4. Given the good quality of the cluster representations obtained we are able to map them easily to their respective newsgroup For instance, cluster 3 represents sci.crypt and cluster 7 summarizes sci.space.

A.4 Summary

In this experimental study we explored pairing different implementations of k-Means that allow it to be used online with different improved distance metrics to reduce the need for large sample batch learning. Supported by both quantitative and qualitative results on social-media streams partitioning, we concluded that NKL and the standard k-Means implementation delivered the best results. k-Means online topical partitioning has a simple setup that is straightforward and contains few parameters, allowing it to be easily implemented in an organization. Thus, we believe that further research in clustering metrics for short text is an area open to new theoretical contributions.