

# **Informative Feature Discovery for Social Media Mining**

Submitted in fulfilment of the requirement for the degree of  
Doctor of Philosophy



**Khaled Mohammed Albishre**

*BCompSc*

School of Computer Science  
Science and Engineering Faculty  
Queensland University of Technology

2020







### **Copyright in Relation to This Thesis**

© Copyright 2020 by Khaled Mohammed Albishre. All rights reserved.

### **Statement of Original Authorship**

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

**Signature:** [QUT Verified Signature](#)

**Date:** 10/04/2020



*To my family*





# Abstract

---

In the web 2.0 era, social media platforms (such as Twitter and Facebook) have become a real-time source of information exchange for millions of people. On these platforms, users can communicate and share their stories. The published data on social media can be overwhelming due to the unprecedented amount of user-generated content. In order to reduce irrelevant and redundant information, an effective social search engine is fundamental to satisfy users' needs with relevant information. By comparing users search behavior on the web, researchers have observed that users tend to describe their information needs with short queries in social media platforms. To improve the retrieval effectiveness of search, relevance feedback is a difficult task requiring large numbers of label documents in the collection, which is not always readily available and can be expensive to obtain in social media data. Thus, to avoid the lack of human-labeled evidence, the aim of this thesis is to discover informative feature representations that can help to capture user information need when no supervised data is available.

Using state-of-the-art techniques in text mining and information retrieval research, this research proposes novel methods to boost the user information need with representative information in a social media context. Firstly, a topic aware of pseudo-relevance feedback is proposed. A Latent Dirichlet Allocation (LDA) topic modeling algorithm is adapted to discover an underlying feature from the initial search results. The concept

of latent topics discovery is used to expose a semantic feature representation. Then, it integrates the lexical relevance evidence with the discovered topical evidence.

This thesis also suggests a method for setting the selection of dynamic feedback documents. It optimises both the robustness and effectiveness for Pseudo Relevance Feedback (PRF) where a certain number of top-ranked documents are assumed to be relevant. The proposed model revises this hypothesis to reduce uncertain information in a high sparseness context, such as social media data in the initial ranked documents. The top-ranked documents set for a given query are produced by sampling a random variable, with its distribution of latent features, using LDA to select the most informative set of pseudo feedback documents.

In addition, an approach is proposed to represent the implicit relationships in the short text from social media. The aim is to address the lack of feature co-occurrences without requiring extra parameters and external evidence. This method transforms the initial ranked documents into a virtual documents space based on the distribution of query terms. To better understand the feature relevance, it weights in terms of the presence of their association in the new space to discover the informative feature.

This thesis conducted substantial experiments using the standard TREC 2011-2014 microblog datasets to evaluate the effectiveness of the proposed models. The experimental results show that the proposed models outperform on all datasets compared to state-of-the-art lexical, temporal, and topic-based retrieval methods. This research lays the foundation for interesting future work that could utilise the proposed models in different aspects of social media mining and information retrieval applications.

# Keywords

---

Information Retrieval

Microblog Retrieval

Text Mining

Topic Modeling

Query Expansion

Relevance Ranking

Social Search

Social Media Mining



# Acknowledgments

---

I praise Allah (God), the most merciful and the most gracious for blessing me with much more than I deserve for every day for everything that happens for undertaking and achieving this research. Without the help of many people, this research project would not have been feasible. Firstly, I would like to express my sincere gratitude to my principal supervisor Professor Yuefeng Li for his constant support, encouragement, and patience throughout this research work. He has been providing me with his excellent guidance and insights with his exceptional knowledge. I am also thankful to my associate supervisor, Associate Professor Yue Xu, for the generous support and advice throughout this candidature. I would also like to thank Professor Yu-Chu Tian and Dr. Jinglan Zhang for examining my Ph.D. thesis.

I am appreciative and grateful to my educational institution in the Kingdom of Saudi Arabia, Umm Al-Qura University (UQU), for giving me the chance to complete my Ph.D. and to provide me with financial support throughout my Ph.D. One cause of my successful graduation is their support during my study journey. I also appreciate the support for logistics, facilitation, and coordination between Umm Al-Qura University and the Queensland University of Technology (QUT) from the Saudi Arabian Culture Missions (SACM) in Australia. I want to express my special thanks to the Science and Engineering Faculty, QUT, which provided me with a supportive research environment with the necessary facilities, including travel benefits, during the period

of my candidature. I also would like to extend my thanks to Ms. Jessica Gregory for her generous work in proofreading my thesis.

There is no way you cannot find thankful and fulfilment for many people. I am profoundly grateful for the tremendous encouragement from my parents. I also want to thank my wife Yaman for her help in these difficult times, with full dedication and compassion. I learned a lot from my beautiful kids, Waleed and Liana, with profound gratitude. Finally, my thanks to my brother and sisters for their continued support.

# Table of Contents

---

<b>Abstract</b>	<b>v</b>
<b>Keywords</b>	<b>vii</b>
<b>Acknowledgments</b>	<b>ix</b>
<b>List of Figures</b>	<b>xix</b>
<b>List of Tables</b>	<b>xx</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview and Motivation . . . . .	1
1.2 Thesis Statement . . . . .	15
1.3 Research Questions . . . . .	16
1.4 Contributions . . . . .	17
1.5 Publications . . . . .	18
1.6 Organisation of the Thesis . . . . .	19
<b>2 Literature Review</b>	<b>21</b>

2.1	Information Retrieval . . . . .	22
2.1.1	Retrieval Models . . . . .	24
2.1.1.1	Boolean Model . . . . .	24
2.1.1.2	Vector Space Model . . . . .	25
2.1.1.3	Probabilistic Models . . . . .	27
2.1.1.4	Language Models . . . . .	27
2.1.2	Temporal Information Retrieval . . . . .	29
2.1.3	Query Expansion . . . . .	31
2.1.3.1	Pseudo Relevance Feedback (PRF) . . . . .	31
2.2	Information Analysis in Social Media . . . . .	33
2.2.1	Overview of Social Media . . . . .	33
2.2.2	Microblog Retrieval Models . . . . .	35
2.2.2.1	Social features based models . . . . .	36
2.2.2.2	Time aware Features Models . . . . .	40
2.2.2.3	Learning to Rank based Models . . . . .	42
2.2.2.4	Fusion based Models . . . . .	43
2.2.3	TREC Microblog Retrieval Tracks . . . . .	45
2.2.4	Microblog Applications . . . . .	46
2.2.4.1	Microblogs Summarisation . . . . .	46
2.2.4.2	Microblog Personalisation . . . . .	50
2.2.4.3	Opinion Retrieval . . . . .	52
2.2.4.4	Topic and Event Detection . . . . .	53



2.2.5	Query Expansion in Microblogs . . . . .	56
2.3	Unsupervised Learning . . . . .	59
2.3.1	Topic Modeling . . . . .	59
2.3.1.1	Topic modeling for social media content . . . . .	60
2.4	Summary . . . . .	64
<b>3</b>	<b>Topic Aware Pseudo-Relevance Feedback for Boosting Microblog Search</b>	<b>67</b>
3.1	Introduction . . . . .	67
3.2	Preliminaries . . . . .	68
3.2.1	Latent Dirichlet Allocation (LDA) . . . . .	68
3.2.2	Language Model . . . . .	69
3.3	Topic aware PRF framework . . . . .	72
3.3.1	Infer topical evidence . . . . .	73
3.3.2	Estimate the relevance feedback . . . . .	75
3.3.3	Integration the evidence . . . . .	76
3.4	Algorithms . . . . .	77
3.5	Summary . . . . .	78
<b>4</b>	<b>Discovery of Informative Training Set for Effective Microblog Search</b>	<b>81</b>
4.1	Introduction . . . . .	81
4.2	The Proposed Model (TBS) Framework . . . . .	84
4.2.1	Pseudo Feedback Selection . . . . .	84
4.2.1.1	Pseudo Documents Construction . . . . .	86

4.2.1.2	Candidate Subset Topical Coverage Estimation . . .	87
4.2.1.3	Representative Candidate Subset Selection . . . . .	88
4.2.2	Expansion Terms Selection . . . . .	89
4.3	Algorithms . . . . .	90
4.4	Summary . . . . .	91
<b>5</b>	<b>Query-based Unsupervised Learning for Improving Social Media Search</b>	<b>93</b>
5.1	Introduction . . . . .	93
5.2	Problem Formulation . . . . .	95
5.3	The Proposed Model (QUSTM) . . . . .	96
5.3.1	Latent Relationships . . . . .	96
5.3.2	Term Estimation . . . . .	100
5.3.3	Algorithms . . . . .	102
5.4	Summary . . . . .	104
<b>6</b>	<b>Experiments</b>	<b>105</b>
6.1	Data Collections . . . . .	106
6.1.1	Preprocessing . . . . .	108
6.2	Experiment Measures . . . . .	111
6.3	Baseline Models . . . . .	115
6.4	Experiment Settings . . . . .	118
6.5	TAPRF Evaluation . . . . .	118
6.5.1	Overall Results . . . . .	118

6.5.2	Discussion . . . . .	121
6.5.2.1	Parameters tuning . . . . .	122
6.5.2.2	Feedbacks Sensitivity . . . . .	123
6.5.2.3	Topics Sensitivity . . . . .	124
6.5.2.4	Per-Query Analysis . . . . .	125
6.6	TBS Evaluation . . . . .	126
6.6.1	Overall Results . . . . .	126
6.6.2	Discussion . . . . .	129
6.6.2.1	Effects for number of feedback documents . . . . .	129
6.6.2.2	Effects for number of expansion terms . . . . .	130
6.6.2.3	Effects of the interpolation parameter . . . . .	130
6.6.2.4	Effects of the topics number . . . . .	131
6.6.2.5	Effects of the subsequence value . . . . .	132
6.6.2.6	Per-query analysis . . . . .	132
6.7	QUSTM Evaluation . . . . .	133
6.7.1	Overall Results . . . . .	133
6.7.2	Discussion . . . . .	137
6.7.2.1	The proposed model sensitivity. . . . .	137
6.7.2.2	Compared with LDA . . . . .	140
6.7.2.3	Per-query analysis. . . . .	141
6.8	Compared Results With TREC . . . . .	141
6.9	Summary . . . . .	145

<b>7</b>	<b>Conclusions</b>	<b>147</b>
7.1	Synthesis of contributions . . . . .	149
7.2	Limitation and Future Directions . . . . .	151
<b>A</b>	<b>TAPRF details results in the TREC microblog dataset</b>	<b>155</b>
<b>B</b>	<b>TBS details results in the TREC microblog dataset</b>	<b>163</b>
<b>C</b>	<b>QUSTM details results in the TREC microblog dataset</b>	<b>171</b>
	<b>References</b>	<b>209</b>

# List of Figures

---

1.1	Internet vs. Social Media Users from 2010 to 2018. . . . .	2
1.2	The frequency distribution of words in the top-30 ranked tweets for query “social media as educational tool”. . . . .	13
1.3	The topics distribution in the top-30 ranked tweets for query “social media as educational tool”. . . . .	14
2.1	Related research areas and coverage of literature review . . . . .	21
2.2	Related research areas and coverage of literature review . . . . .	23
2.3	An Example of Tweet. . . . .	35
3.1	The proposed model framework. . . . .	72
3.2	Topic Example for the top 50 documents in MB2011 . . . . .	74
4.1	TBS general architecture . . . . .	85
4.2	Selecting pseudo-relevance feedback . . . . .	86
5.1	The relationship between $Q$ , $\Omega$ and intermediate sets. . . . .	99
5.2	The relationship between $\Omega$ and $Q$ via $R$ . . . . .	100

6.1	An example of tweet structure . . . . .	108
6.2	An example of query in the test set . . . . .	108
6.3	relevanc Judgment Ratio . . . . .	109
6.4	TAPRF parameters sensitivity. . . . .	122
6.5	TAPRF number of pseudo-feedbacks sensitivity. . . . .	124
6.6	TAPRF number of topics sensitivity. . . . .	125
6.7	TAPRF per-query analysis. . . . .	126
6.8	Sensitivity of the number of feedback documents $\Omega$ in (a) and the number of expansion terms in (b) on TBS for the microblog TREC 2011 collection. . . . .	130
6.9	Sensitivity of the number of interpolation parameters $\lambda$ in (a) and the number of topics $V$ in (b) on TBS for the microblog TREC 2011 collection. . . . .	131
6.10	Sensitivity of the sequence value $x_j$ on TBS at microblog TREC 2011 collection. . . . .	132
6.11	Difference in MAP between TBS and QL using the MB2011-MB2012 topic sets in (a) and the MB2013 and MB2014 topic sets in (b). . . . .	133
6.12	The proposed model parameters sensitivity . . . . .	137
6.13	The proposed model performance in terms from a selected number of tweets $T$ and terms $Q'$ for all test sets. (a) MB11 and (b) MB12 . . . . .	138
6.14	The proposed model performance in terms from a selected number of tweets $T$ and terms $Q'$ for all test sets. (a) MB13 and (b) MB14. . . . .	139
6.15	Difference in the P30 and MAP between the proposed model and QL across all test sets. . . . .	142

## List of Tables

---

1.1	Top results for query “social media as educational tool”.	12
4.1	An Example of subset constrain	86
5.1	An example of virtual documents.	97
6.1	The statistics of test collections	107
6.2	Contingency Table	112
6.3	Comparison of the proposed method TAPRF and baselines lexical based models.	119
6.4	Example of expanded terms for a topic numbered MB86: “Michelle Obama’s obesity campaign”.	121
6.5	Comparison of the proposed method TBS and baselines models over MB2011 and MB2012 test sets.	127
6.6	Comparison of the proposed method TBS and baselines models over MB2013 and MB2014 test sets.	128
6.7	Comparison of the proposed method QUSTM and baselines models over MB2011 and MB2012 test sets.	134

6.8	Comparison of the proposed method QUSTM and baselines models over MB2013 and MB2014 test sets. . . . .	135
6.9	Results comparison. . . . .	140
6.10	The performance comparison of the proposed models TAPRF, TBS and QUSTM with the submitted TREC's (2011-2012) microblog track runs. . . . .	144
6.11	The performance comparison of the proposed models TAPRF, TBS and QUSTM with the submitted TREC's (2013-2014) microblog track runs. . . . .	145
A.1	Details TAPRF Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014 . . . . .	155
B.1	Details TBS Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014 . . . . .	163
C.1	Details QUSTM Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014 . . . . .	171



# Chapter 1

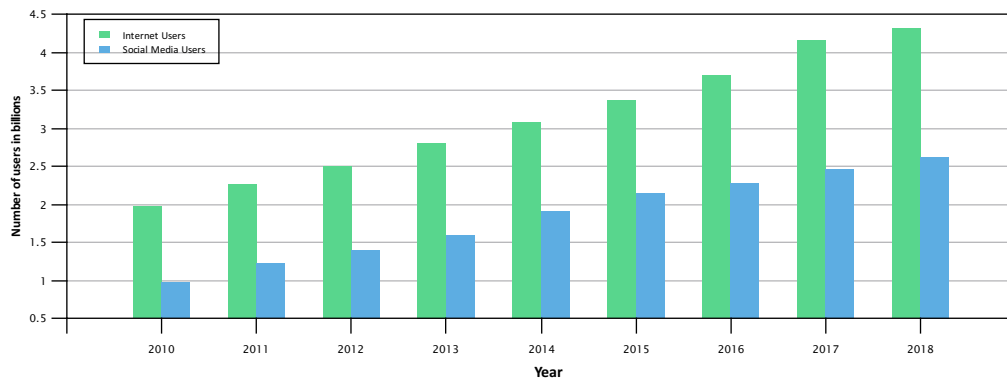
## Introduction

---

### 1.1 Overview and Motivation

The World Wide Web (WWW) has significantly transformed the way information is consumed, produced, processed, and gathered. In the last few decades, Web 2.0 has become a virtual world where users have not only been consuming information but also creating data. Social media has experienced a boom in recent years, with developments in portable smart devices allowing users to express themselves freely, via various mediums, and without any barriers. Social media mining aims to use data from user-generated content to develop knowledge decision-making models that can make the user experience of exchanging information easier. Different applications can extend such knowledge, including e-commerce recommendation systems, personalisation of news feeds, and content analysis. Two specific types of information are generated from social media: graph or network data and text data. Working with text data is the main focus of this research.

Online social media platforms are recognised as websites that allow individuals to generate, share or exchange data. Wikis, blogs, microblogs, social networking, media



**Figure 1.1:** Internet vs. Social Media Users from 2010 to 2018.

sharing, and social bookmarking represent a wide variety of social media. The most comprehensive social media platforms include Facebook<sup>1</sup>-2004, Youtube<sup>2</sup>-2005, Twitter<sup>3</sup>-2006, LinkedIn<sup>4</sup>-2006, and Instagram<sup>5</sup>-2010. These platforms enable interaction and participation from users who belong to the virtual community. The number of users on these platforms continues to increase in popularity daily across the globe. Figure 1.1 shows the number of social media users worldwide compared to Internet users in the period 2010 to 2018. Statistics clearly show the number of social media users is growing almost linearly, with an average of about 2.77% users per year and approximately 62% of all Internet users using social media platforms in 2018. Such evidence motivates researchers and the technology industry to develop sophisticated methods that can extract useful information from social media data to address users' immediate needs.

The rapid information sharing made possible via social media creates an incredible

---

<sup>1</sup><https://www.facebook.com/>

<sup>2</sup><https://www.youtube.com/>

<sup>3</sup><https://www.twitter.com/>

<sup>4</sup><https://www.linkedin.com/>

<sup>5</sup><https://www.instagram.com/>

information source, offering real-time news compared to slower traditional mediums. However, in terms of redundant and/or irrelevant information, the streams on social media can easily overwhelm users. This is magnified by a growing volume of fake news, and malicious content that continues to grow as result of free and easy access to user-generated content (UGC). Contrarily, traditional sources of news media (i.e., professional organisation) continues to offer reliable content after an event. On the other hand, social media can offer real-time coverage that that may misrepresent what is really happening. Hence, both the advantages and downsides of this development are noted.

Twitter is one of the most important social media platforms. It allows people to communicate with each other by publishing and sharing their ideas and opinions. A user can produce a short text post, called a tweet, that is limited to only 140 characters. Statistics show that Twitter has 326 million active monthly users and 100 million active daily users, sending 500 million tweets a daily<sup>6</sup>. A user may be interested in the content of other users and may follow them to receive their posts. Users receive updates from the users they follow in their timeline. Users can then interact with incoming feeds by republishing tweets to their followers (a retweet) or by liking. Users on Twitter can use a variety of network features, including tweets to a specific topic using hashtag keywords (e.g., #QUT).

Due to the timeliness and variety of user-generated content on social media, platforms have become a crucial real-time source of news for ongoing events. Different events on Twitter have shown that microblog sites have become a key channel for exchanging data between people, when compared to traditional media. For instance, the death of Michael Jackson was confirmed on Twitter before reaching news media [Kaplan and Haenlein, 2011]. Many organisations (such as NASA) have opted to post

---

<sup>6</sup><https://www.omnicoreagency.com/twitter-statistics/>

substantial stories on social media platforms<sup>7</sup>. Recently, U.S. President Donald Trump has utilised his Twitter account to announce significant news. The power of social media platforms comes from their ability to provide widespread information access to mass users.

The three major benefits of social media are coverage, freshness, and timeliness. Coverage refers to the breadth of topics, issues, personal stories, and more, covered via UGC on social media. Due to rapid real-time production, the content is also far fresher on social media than on traditional media. Content may cover everything from live sports events to natural calamities and other valuable information. Social media has thus become a significant real-time source of information, particularly during uncommon events or emergencies such as floods, terrorist attacks, cyclones, earthquake or social resistance. For example, Sakaki et al. [2010] conclude that Twitter has helped to detect nearly 75% of earthquakes within two minutes through the monitoring of “earthquake” and related terms. Timeliness is evident in specific timestamps embedded in a social media message. These stamps play a pivotal role in tracking a topic and the development of content over time. Compared to traditional corpus/archives (static), text in the social media is much different. In social text streams, both viewpoint and topic drifts may be noticeable. Due to these features, over time, the statistical features of social media texts can change.

Despite these advantages, social media struggles with five significant disadvantages: volume, velocity, variety, redundancy, and quality. It is now commonplace for many users to use social media to share information daily. This is in comparison to traditional media where there is no restriction for publishing quantity. Thus, it may be seriously overwhelming to deal with the extensive information published on social media. Secondly, the rate with which social media content is generated is

---

<sup>7</sup><https://www.wired.com/2015/07/nasas-social-media-strategy-genius-kinda-maddening/>

unprecedented. This velocity is aided by simple user interfaces. Thirdly, a variety of topics are covered by social media posts that may range from commonplace chatter to historical events at a global scale. Also, contrary to social media, topic drift is common as messages are not classified. Likewise, the same event may be discussed by a large number of users in different languages. Fourthly, overwhelming redundancy is a clear aspect of social media. Though it may be effective for swiftly of noteworthy occurrences, social media may also add enormous redundancy as many people may be share or publish the same information across different points in time. Finally, since the primary source of information on social media is the user, when compared to traditional news sources, and there are no ‘in-principle’ checks on messages, the quality of the content can be called into questions. This leads us to the two major areas to consider in regards to information quality: the focus placed by a user and the quality which is mostly carried in colloquial language and expression.

Social media platforms often have two types of primary users: content seekers and content producers [Java et al., 2007]. Based on information seeking behaviors, Efron [2011] defined two types of microblog search users: users “asking for information” and “retrieving information”. When compared to research problems related to Information Retrieval (IR), these two user groups are equivalent to Q&A task and ad-hoc search. Content producers indicate users who share or write content on a given topic or interest, such as trending events or emerging news. This thesis concerns users who are looking for relevant tweets based on their information needs.

A massive quantity of user queries seeking relevant information have been submitted to social media platforms. Over 2 billion search queries<sup>8</sup> are sent to Twitter every day. Twitter includes a search interface for users to enter a set of terms and retrieve tweets in reverse chronological order; however, Twitter generates a high volume of data

---

<sup>8</sup><https://blog.hootsuite.com/twitter-statistics/>

every second from users, making it extremely difficult to discover relevant information that meets the user's needs. As users overload the search results, relevant tweets could quickly be buried leaving users to think their search query has returned no relevant tweets.

Search engines are the most significant medium supporting daily information access for many users. Text-based search engines support queries directly and can be expanded to provide search results (for users to navigate) or provide recommendations. The ad-hoc retrieval task is the main methodology for accessing IR. The core idea behind this task is to return items that are relevant to an immediate user information need, represented as a set of terms called a query. From the viewpoint of the user, IR is the issue with using a query to locate appropriate items in a collection. This is often a challenge as users often have temporary ad-hoc requirements and want to discover the immediate relevant information.

The effectiveness of finding relevant information in social media retrieval systems can be strongly determined based on the specificity of users' needs and the collections. Many reasons can be involved in why meeting user needs or particular text retrieval in social media data is challenging. Firstly, users in social media platforms tend to express their need in short queries. User queries submitted to Twitter's search engine have been observed to be considerably shorter than those presented to web search engines (i.e., 1.64 words versus 3.08 words) [Teevan et al., 2011]. Secondly, the user does not have enough background information about what she/he is looking for in their query. In this situation, the user information need can be hard to accurately define. Finally, due to the short nature of social media documents, it can be difficult to precisely define their meaning and relevance. The right answer can be subjective; even when human experts assess the relevance of documents, they can disagree with each other.

Obtaining high-quality queries from all users is an infeasible task, so it is essential to re-formulate these queries automatically to satisfy users' needs. Methods of Query Expansion (QE) are introduced to address this problem. Original user queries have been enhanced with various techniques, including global or local document collection analysis [Carpineto et al., 2001, Carpineto and Romano, 2012]. QE can primarily be performed through interactive methods, including Relevance Feedback (RF), Word Sense Disambiguation (WSD) and clustering of search results. Therefore, the aim of QE is to use mining words (with particular relations to the original query words) to extend the original search.

It has been shown that automatic query expansion (AQE) methods are effective. Similarly, many successful AQE methods have been developed in the context of microblog retrieval [Lau et al., 2011, Whiting et al., 2012]. In addition, during TREC Microblog, the majority of participants used the AQE technique incessantly to execute their systems for ad-hoc retrieval. These participants [Ounis et al., 2011, Soboroff et al., 2013, 2014, 2012] reported notable improvements in the retrieval process. Nonetheless, it was also reported that some topics may experience a performance deficiency under the same AQE methods, although the performance of the retrieval may improve on average. Though all of these techniques worked well, most of them depended directly on the score generated by the retrieval model to enhance the terms of query expansion. Retrieval models can be unreliable when working with microblog conditions, so these scores may produce misleading results [Rodriguez Perez and Jose, 2015]. Therefore, a number of stand-alone alternatives have also been proposed in this thesis.

RF has been proven to be an effective technique to increase IR system performance. RF's main objective is to enable the retrieval system to learn from the feedback provided by a user to improve the search results. RF relies on a feedback set that includes

both positive and negative samples. It needs an effective features extraction model to discover relevant features from the feedback set. Text feature selection is a basic method of selecting a subset of features from a set of documents aimed at reducing irrelevant and redundant information [Gheyas and Smith, 2010]. Due to the high sparseness and velocity of incoming short text documents on social media platforms, it is challenging to guarantee the ability of discovered relevance features to accurately represent user requirements. Thus, it is a significant challenge in IR research to develop an optimal retrieval method from both a theoretical and empirical viewpoint.

There are many possible ways to obtain feedback documents. Typically, some methods use sources of explicit evidence (like labeled documents by real users), whereas some utilise implicit evidence (like user clicks data). Feedback information requires additional efforts (such as, an actual user judgment), which is often costly. Users are often not willing to perform additional work when searching, such as to annotate whether or not a document is relevant. In fact, the feedback information is not necessarily available for each particular query. For a specific query, there may not be enough user data, especially for specific topics (such as an emerging disaster).

In the absence of feedback documents, the effective QE technique is Pseudo-Relevant Feedback (PRF) that automates the manual part of the relevance feedback [Buckley et al., 1995, Lynam et al., 2004]. As a basis for selecting the most relevant response to the query, QE via PRF assumes that a proportion of the top-ranked documents (i.e., first-pass retrieval) are relevant to the initial user query. PRF compares feedback words included in feedback documents with query words. Thus, the aim of PRF is to discover the co-occurrence of pseudo-feedback documents and query words in order to identify and expand the related words with the original user query. Some relevant documents that have been missed during the first-pass round can be acquired using these methods in order to improve overall performance and satisfy the needs of



the user. As mentioned earlier, QE research via PRF is crucial and helpful in those IR systems where user judgments are often not available.

Many microblog retrieval studies have used pseudo relevance feedback (PRF) for query expansion. These studies also assume that making use of the most related available terms in those pseudo-relevant documents is valuable. An earlier study proposed in Miyanishi et al. [2013] elects to manually select tweets in the first stage and then estimate the relevant feedback for the selected tweets. In Lin et al. [2012], a graph-based model was introduced that could generate a storyline for a particular query within the PRF framework. To highlight the short-term importance of a given query, Albakour et al. [2013] used PRF to extend user information to capture the sparsity challenge. Alternatively, global knowledge bases (including Freebase or Probase) can be used to bridge the semantic similarity gap in microblogging [Wang et al., 2017]. In Fan et al. [2015], detecting the underlying entities in the original query and then using the relevant feedback model. However, the use of global evidence, such as a knowledge base parallel to the PRF framework, requires a double run for a query, which can decrease the computational efficiency [Carpineto and Romano, 2012]. Most of the above contributions hold the same classic PRF assumption that the retrieved initially documents are relevant to the original query.

Over the last century, many unique retrieval models have been proposed for finding relevant information. A retrieval model can be described as a logical framework that processes the representation of a user query and the documents in a collection to decide the relevance of each document and then rank them based on their relevance to the user's need. Various IR frameworks have been proposed in the existing literature, including the Boolean model, Vector Space Model (VSM) and probabilistic model. These models will be described in more detail in Section 2.1.1. The commonality between these models is the ranking function that aims to assign scores to documents

in relation to a given user need. How representative the retrieval model is of a given feature is a critical determinant of the weight of a feature produced by the ranking function. Notably, not all extracted features are relevant to the learning process. Many are often redundant, unrelated, and sometimes even noise, which can lead to adverse effects such as low retrieval performance [Cai et al., 2018, Qian and Zhai, 2013]. The challenge becomes refining the process of matching the query to sort the return documents based on a criterion that represents the best result.













While a range of social media mining applications try to capture appropriate data, the retrieval model should consider an extensive text representation mechanism to address the needs of the user in an effective and efficient manner. To date, robust IR models have been developed that can be broadly sorted into various key categories based on their features. These include term-based, topic-based, concept-based, and pattern-based models. The majority of content feature extraction methods use term-based models, such as TF-IDF, Okapi BM25 [Robertson et al., 1994], and the language model [Lavrenko and Croft, 2001]. Documents are viewed as a bag of words (BoW) in term-based models, which can only use lexical evidence. The method loses the contextual relationship between extracted features (terms) due to the term independence hypothesis [Wang et al., 2008, Zhai and Massung, 2016]. It is also prone to the use of word variation in a document that is prevalent in social media data. This issue exacerbates the problems of polysemy (different meanings for one word), synonymy (the same meaning for different words) and hyponymy (a word included within other words on the same semantic level). However, the key benefits of term-based methods are efficient computing and the maturity of the term scoring function. Such methods could, therefore, extract noisy and redundant text features that reduce the effectiveness of the retrieval system.

State-of-the-art retrieval models tend to rely on evidence that points to heuristic

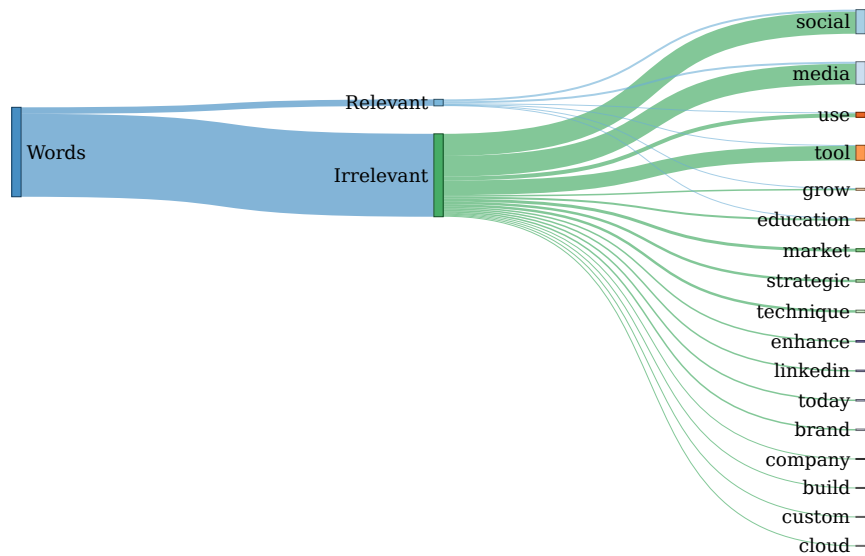
techniques. These factors include the term frequency (TF), document length, and document frequency. The ranking function can use this information to calculate the total weight of a word. These calculations are often combined, in various ways, in the developed retrieval models. Table 1.1 illustrates the top-30 ranked tweets using the query likelihood retrieval model for the query “Social Media as Educational Tool”. Data is taken from the TREC Microblog 2013 dataset based on the premise that user judgments are accessible. Figure 1.1 demonstrates that heuristic evidence, such as word frequency, is part of the retrieval model. As shown in the figure, the words “social”, “media” and “tool” have a relatively high frequency in both positive and negative tweets. Data sparsity is a common issue in short text documents, as their constricted lengths means they provide insufficient contextual information. The BoW model ignores the order and semantic associations between features; therefore, general techniques based on the BoW may not be applied directly in short text documents [Sriram et al., 2010]. The lack of initial user judgments also makes it difficult to capture the relevant information. Thus, modeling of the associations between words in short text documents is crucial for the identification or indication of a word that is important to the user’s needs.

Relevant information in a document can also be generally correlated with information in an irrelevant document. Term-based feature retrieval techniques have faced challenges in identifying the relevance scale of a word from surrounded words. The challenge is magnified by the addition of information from the user or external resources, such as knowledge bases. Researchers have recently started using topic models for discovering relevant information without human supervision to replace feedback documents [Andrzejewski and Buttler, 2011, Jian et al., 2016, Serizawa and Kobayashi, 2013, Yi and Allan, 2009]. The topic model technique assumes there is an underlying semantic high-level representation, called latent topics, to reduce

**Table 1.1:** Top results for query “social media as educational tool”.

Rank	Tweet	Relevant
1	Without <b>social media</b> the internet is an <b>education tool</b> .	
2	<b>Social Media</b> is now an essential part of any brand. We direct <b>educate</b> users on how to use this <b>tool</b> to the best of its ability. #twitter	
3	Using Hashtags as Strategic Objects - <b>Social media</b> news, strategy, <b>tools</b> , and techniques — <b>Social Media</b> Today.. [URL]	
4	Time for <b>Social Media Education</b> . Award Winning <b>Social Media</b> Speaker Offers <b>Social Media Education</b> to Corporations [URL]	
.	...	.
.	...	.
.	...	.
18	PBN panelists: Think about <b>social media</b> as more than a marketing <b>tool</b> [URL]	
19	<b>Social media</b> as <b>tool</b> of e-conversation NOT to build profit or increase bottom line [URL]	
20	<b>Social Media</b> , PR, Branding As A <b>Tool</b> For Changing Perceptions and Aiding the Development of Africa #smwmarketing [URL]	
.	...	.
.	...	.
22	Should Law Schools Be Making Better Use of <b>Social Media</b> as a Teaching <b>Tool</b> ? [URL] #education #law	
23	LinkedIn Replaces Facebook as Top <b>Social Media Tool</b> Among Inc. 500 (Fastest Growing Private Companies) [URL]	
.	...	.
.	...	.
25	<b>Social Media</b> Lovers: Do You Need to Get Engaged? - <b>Social media</b> news, strategy, <b>tools</b> , and techniques — <b>Social Media</b> .. [URL]	
26	<b>Social Media</b> What use is <b>social media</b> in <b>education</b> ?: In view of the growing demand for <b>social media</b> skills, st... [URL]	
.	...	.
.	...	.
30	Execs board members see <b>social media</b> as a marketing <b>tool</b> not worthy of their attention. Hear the story on FIR 691: [URL]	

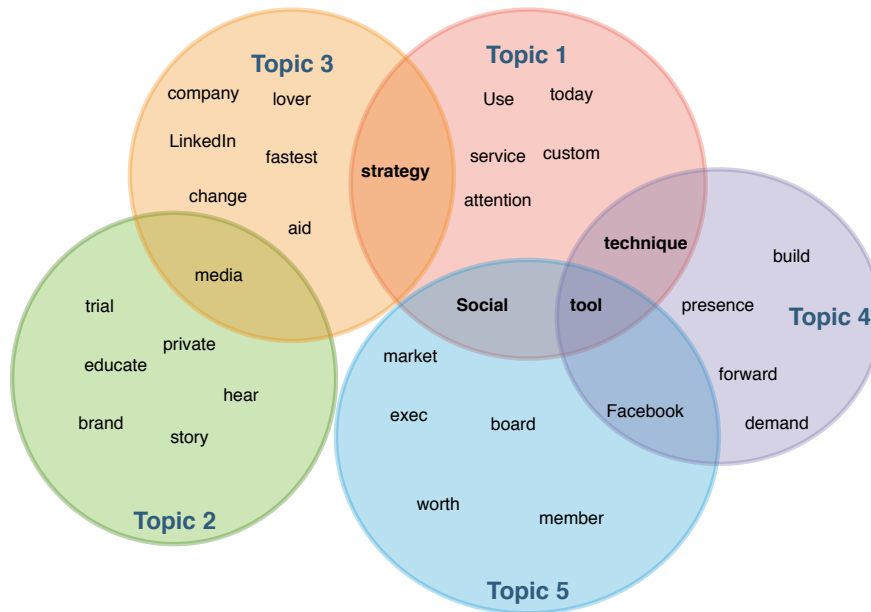
the features appearing in text documents from high to low dimensional. The topic model has been one of the most effective unsupervised learning techniques and has been rapidly accepted in machine learning and natural language processing research communities. A variety of topic models have been deployed in previous studies, including Probabilistic Latent Semantic Analysis (pLSA) [Hofmann, 1999], Latent Dirichlet Allocation (LDA) [Blei et al., 2003], Biterm Topic Model (BTM) [Yan et al., 2013] and the dual-sparse topic model (DSTM) [Lin et al., 2014]. LDA is a generative probabilistic model for a set of documents, presented as mixtures of latent topics, in which the word distribution defines each topic (more details shown in Section 2.3.1). As shown in Figure 1.1, we illustrate the topic distributions for the top-30 ranked tweets. It is demonstrated that LDA has the ability to infer the associations across terms in the form of interrelated topics, that can be useful to discover relevant information.



**Figure 1.2:** The frequency distribution of words in the top-30 ranked tweets for query “social media as educational tool”.

LDA has become one of the most popular probabilistic text modeling techniques and has been quickly adopted by machine learning and text mining communities.

While the retrieval performance improved significantly using PRF based methods, the classical PRF based models have also shown limitations with short documents, such as tweets. For instance, the query performance was suggested by a broad spectrum of predictors, generally connected to the performance of PRF [Hauff, 2010, He and Ounis, 2009]. All these past studies have concluded that PRF effectiveness is critical for the quality of feedback. As feedback documents are not evaluated by real users when using PRF, there is no guarantee of the quality of a set of feedback documents. A feedback document in a particular query topic may not be useful even if it is found to be relevant. The given document may only be partially relevant, with a small section covering the query topic while the rest of the document is irrelevant. In this case, the query is augmented by insignificant expansion words, resulting in decreased retrieval performance. To put it another way, term occurrence relations are



**Figure 1.3:** The topics distribution in the top-30 ranked tweets for query “social media as educational tool”.

not a competitive choice for term selection when the term space lacks enough relevant information. Relevance is, therefore, not sufficient to determine whether the document can help PRF in these scenarios to improve their performance. In order to evaluate a feedback document, this research uses “informative” instead of “relevance”. The concept of “informative” in this thesis differs from how we consider a query as already mentioned. A quality feedback document is particularly important and useful for improving the final performance of PRF. Thus, the selection of good quality feedback documents Ye et al. [2013] is crucial. Furthermore, it has not been determined how to define an “informative” feature discovery in the context of short documents in social media data.

In summary, the selection of text features has become an essential component of

social media mining. Researchers have faced challenges in determining relevant text features in the absence of relevant feedback information and, due to the nature of social media text, it is an ongoing research problem. The important role of text feature selection is not only to discover features but to find the most informative features in order to meet the user's information need.

## 1.2 Thesis Statement

This research thesis focuses on developing methods to address the challenges posed by microblog retrieval. First, we are studying the applicability of introducing topical distribution in the relevant model. We hypothesise that a set of first-retrieved microblog posts has latent relationships where it reflects part of the relevance of microblog documents to a given user query. While the latent relationships are considered as topical evidence, the relevance model can be used to discover lexical evidence. We are thus contributing a new topic-conscious pseudo-relevance feedback model (TAPRF) that significantly captures the relevance of a two-level microblog method.

In the absence of any human relevance judgments, selecting a training set from the initially retrieved documents without considering their quality may introduce more noise, especially with applications such as microblog retrieval. In turn, this reduces a feedback model's ability to capture information relevant to a user's needs, making the determination of the informative training set for relevant feedback without extra effort from the user a critical challenge. Where the ratio of relevant to irrelevant documents is unknown, we assume that the terms set out in the relevant documents are semantically related in latent relationships but are diverse in the irrelevant documents. To address this issue and improve the performance of short text document search, we designed an innovative two-step mechanism to automatically select a set of pseudo-documents

from the first passed documents for a given user query. The aim of this model is to discover those latent topics in the top-ranked documents that enable the correlation between terms in relevant topics to be exploited. To capture discriminative terms for query expansion, we incorporated topical features into a relevant model that focuses on temporal information in the selected set of documents.

The quality of extracted features from a set of tweets depends heavily on how adequately the user formulates their needs. This is not always available. As mentioned earlier, the nature of tweets (short in length and sparse) is challenging and provides insufficient correlations between terms to reflect discriminative power. We assume that relationships of relevance can be determined if we can strengthen terms in a set of tweets. To reach this assumption, we introduce a new query-based aggregation scheme that accumulates a set of tweets in the initially retrieved documents, including a query term in a single virtual document. Therefore, a set of virtual documents will be introduced where a tweet may belong to more than one virtual document based on the inclusion of query terms. This mechanism bridges the gap introduced by the lack of word co-occurrences without requiring the estimation of many parameters or the collection of external evidence. To reflect the implicit relationships for a term on the new space, we model the association between extracted terms using a proposed weighting function where it is considered to behave across virtual documents.

### **1.3 Research Questions**

This research focuses on the problem of improving the performance of a microblog search for a given user query by finding informative features to represent user information needs. With regard to the search for microblogs, this thesis has organised several main research questions. Guide to the following research question in Chapter



3:

**RQ1:** Is the unsupervised learning model-topic modeling reliable to infer high-quality terms from unlabeled tweets (e.g., a set of pseudo-documents) to meet a user's need?

Subsequently, Chapter 4 further explores the selection of the dynamic set of pseudo-documents and provides the following research questions:

**RQ2:** How can we adapt the topic modeling to discover representative pseudo tests for a given user query?

Due to the high sparsity of the initially retrieved documents set, Chapter 5 raises the uncertainty of incoming pseudo-documents to infer the relevant informative features.

**RQ3:** How can we determine the importance of a feature in a set of pseudo-documents without relying on human effort (e.g., judgments of relevant documents)?

**RQ3.1:** Can we build an augmentation scheme for the first-pass documents retrieved based on their association with the original query terms?

**RQ3.2:** How can we model the relationships between terms to reflect the discriminative power of these terms?

## 1.4 Contributions

In this section, we are summarising the contributions of this research and mapping them to the research question.

- C1:** A study of what makes the extracted topical terms set from a set of pseudo feedback using the topic model technique, such as the Latent Dirichlet Allocation (LDA) are representative to find user query related terms. The experimental results of this study are significantly improved compared to the lexical-based retrieval models (Related to **RQ1**).
- C2:** To reduce unreliable information in the initial retrieved documents set to improve microblog search, we propose an automatic topic-based model which can select the training set before applying PRF based on the topical distribution of features in the initially retrieved documents. Then, we integrate the topical features of the selected training set with lexical evidence from the relevant feedback model to identify the expanding features set. Also, we consider the temporal distribution of each document from selected pseudo-documents (Related to **RQ2**).
- C3:** To address data sparsity in initially retrieved tweets, we aggregate tweets based on query-based pooling. Then, we describe the relationships between data items (i.e., original query terms, virtual documents and terms) to reduce information uncertainties in the proposed aggregation technique. From the discovered relationships, we propose a weighting scheme that can estimate the appropriate score for each term to reflect its discriminatory power (Related to **RQ3**).

## 1.5 Publications

- [Albishre et al., 2015] Albishre, Khaled, Mubarak Albathan, and Yuefeng Li. “Effective 20 newsgroups dataset cleaning.” In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*. 2015.
- [Albishre et al., 2017] Albishre, Khaled, Yuefeng Li, and Yue Xu. “Effective

pseudo-relevance for microblog retrieval.” In *Proceedings of the Australasian Computer Science Week Multiconference*. 2017.

- [Albishre et al., 2018] Albishre, Khaled, Yuefeng Li, and Yue Xu. “Query-Based Automatic Training Set Selection for Microblog Retrieval.” In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. 2018.
- Khaled Albishre, Yuefeng Li, and Yue Xu. “Discovery of Informative Training Set for Effective Microblog Search.” *Artificial Intelligence Review*.2019. (Submitted to Journal)
- [Albishre et al., 2019] Khaled Albishre, Yuefeng Li, Yue Xu, and Wei Huang. “Query-based Unsupervised Learning for Improving Social Media Search”. *World Wide Web*. November, 2019.

## 1.6 Organisation of the Thesis

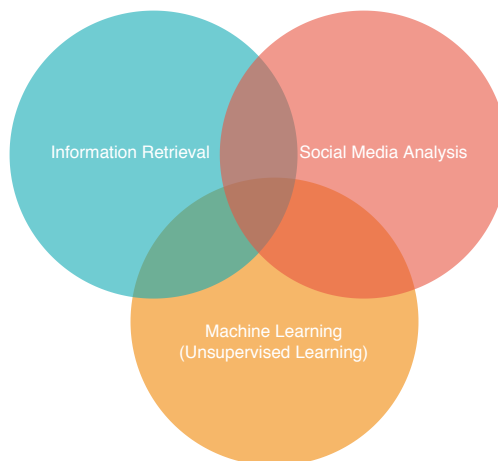
- **Chapter 2:** This chapter is a literature review of the concepts and backgrounds that will be used in this research, including information recovery (IR), information analysis in social media, and unsupervised machine learning. It comprehensively reviews the fundamentals of related IR techniques and identifies the drawbacks when using social media data. The state-of-the-art methodologies and applications for finding relevant information on social media data are reviewed. Finally, topic modeling developments with microblog posts are discussed.
- **Chapter 3:** This chapter describes, in detail, the topic aware pseudo relevance feedback (TAPRF) model. First, it presents the general framework for the proposed model. It then details the components of the framework, including topical evidence and estimation of the relevant model.

- **Chapter 4:** This chapter presents the proposed model (TBS) for the dynamic selection of pseudo-feedbacks set for a given user query. First, it describes the proposed method, which contains two main steps. It investigates the selection mechanism in the first phase based on the distribution of the topical words in a set of initially ranked tweets. It then shows how the selected tweets are integrated with the relevant model.
- **Chapter 5:** This chapter presents a novel query-based unsupervised learning method called Query-based Unsupervised Short Text Mining (QUSTM) to improve the effectiveness of social media search. To obtain high-quality search results from the mass of social media data, QUSTM represents the implicit relationships in short texts and aims at addressing the lack of word co-occurrences without requiring extra parameters and external evidence.
- **Chapter 6:** This chapter describes benchmark collections and performance metrics as well as the application of the proposed models for microblog retrieval applications. It also presents a detailed evaluation analysis compared to state-of-the-art baseline models.
- **Chapter 7:** This chapter concludes and summarises this research thesis, highlighting contributions and proposing research in future directions.

## Chapter 2

### Literature Review

---



**Figure 2.1:** Related research areas and coverage of literature review

In this chapter, we present the state-of-the-art concepts related to our research. We start with a general introduction of Information Retrieval (IR) in Section 2.1, which focuses on related information retrieval concepts in social media research. We address IR in this thesis from three directions: retrieval models in Section 2.1.1, query expansion techniques in Section 2.1.3, and temporal IR in Section 2.1.2). Afterwards, in Section 2.2, we discuss information analysis in a social media context starting with

an overview of social media that includes the benefits and challenges. We review the related work on microblog retrieval approaches from different technique's perspectives, including social, time-aware, and query-based features. To account for recent expansion in microblog application, Section 2.2.4 recalls previous microblog directions, including summarisation, and event and topic detection. Finally, in Section 2.3, we discuss related work in machine learning, including topic modeling and correlations with IR research.

## 2.1 Information Retrieval

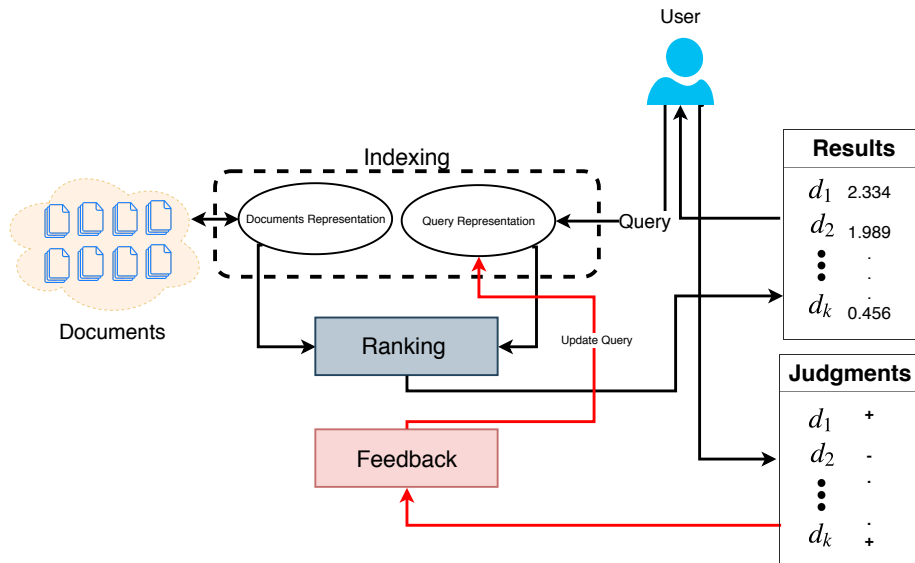
Information retrieval (IR) is the fundamental line of research in computer science concerned with helping a user find relevant information in data collection. At present, many successful information retrieval applications (such as search engines like Google<sup>1</sup> or Bing<sup>2</sup>) are used in our daily routines. Different definitions have been proposed to characterise these information retrieval concepts in the literature. According to Manning et al. [2008] definition "Information Retrieval is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)". Thus, IR systems return a set of relevant information from a collection to satisfy the user's need.

The typical IR system consists of three main steps to fulfil a user's information need, commonly represented by a set of terms called a query. These steps are: documents indexes, processing user queries, and matching processes, as show in Figure 2.2. The indexing phase is an offline process that is performed at the begin of the IR system cycle. The main purpose of the indexing stage is to ensure efficient mapping

---

<sup>1</sup><https://www.google.com/>

<sup>2</sup><https://www.bing.com/>



**Figure 2.2:** Related research areas and coverage of literature review

among documents and terms where they happen. The most common index structure is an inverted index, consisting of a dictionary and posting lists. The inverted index is storing a mapping from document content, such as terms, to its locations in a document or a set of documents. Indexing text documents in a collection involves several important preprocessing steps including, but not limited to, normalisation, tokenising, stemming, and stop word removal. Processing user information need is the next IR system step. Regularly, the user expresses a few terms to an IR system to find relevant information. This step is called query formulation. The critical step in an IR system is matching documents that are relative to the user query. In this step, the system returns a set of ranked documents in descending order based on the similarity

between documents and a query. This is known as document relevance. An IR system is required to estimate the relevance between documents and a query. The system then ranks these documents based on its score. Different types of algorithms have been proposed in previous studies, as shown in Section 2.1.1.

## **2.1.1 Retrieval Models**

### **2.1.1.1 Boolean Model**

The Boolean Model is the first of several IR models proposed by Lancaster and Fayen [1973], Van Rijsbergen [1979]. The conceptual framework of the Boolean Model is based on a set theory where each document is denoted as a binary set of its contained terms. Term significance, query, document or collection are not assigned any notation. A composing query with Boolean logic operators (such as AND, NOT, and OR) is utilised to require or exclude the occurrence of query terms in relevant information to express relevant information. Outcomes represent a binary decision that is TRUE or FALSE, since it depends on logical conditions. Thus, as there is no scale of relevance estimation, the Boolean Model's results are given as an unranked set. Document features (such as gender) can be utilised by a query to encourage a result ranking.

Several challenges are posed to the traditional retrieval model based on Boolean logic. First, when the user's information need is involved, queries based on Boolean logic are a complicated and relatively unnatural means to demonstrate uncertain information needs. Boolean queries can also be impractical, especially in more extensive data collections due to the many conjunctions and disjunctions that are required to satisfy the user's information need. This way can reduce the search result and could improve the overall precision.



### 2.1.1.2 Vector Space Model

The Vector Space Model (VSM) is a simple, yet effective technique of scheming ranking purposes for information retrieval systems. Salton et al. [1975] propose VSM to address the limitations of the Boolean Model. VSM reinforces partial matching and integrates relevance estimation in result ranking. It represents documents and queries as vectors in a multi-dimensional space wherein relevance estimation is defined as the space between vectors. This kind of representation is based on Euclidean geometry. The VSM was proposed because it affords a better fit to the more intricate IR techniques and an intuitive elucidation of the problems of IR, such as relevance feedback and term estimations that prompt retrieval effectiveness [Croft et al., 2015].

The main idea behind VSM is genuinely straightforward to understand. The VSM framework makes a set of assumptions. The first assumption is that were present each document  $d_i = \{t_{1,i}, t_{2,i}, \dots, t_{n,i}\}$  and query  $q = \{t_{1,q}, t_{2,q}, \dots, t_{n,q}\}$  through a term vector representation. At this point, a term that is assumed to express one dimension represents any basic notion, such as a word or a phrase,  $n$ -grams or any other feature representation. The VSM defines a  $|V|$ -dimensional space, since we have  $|V|$  terms in our vocabulary list. To emphasize term discriminative, the VSM employs term weighting schemes, such as term frequency (TF) or inverse document frequency (IDF), to the weights of different terms. The relevance, in this case, is measured based on the similarity between the query vector and document vector.

Several heuristics ranking functions of term weighting have been examined with a VSM framework. TF is the most straightforward way to express the count of term  $t$  in document  $d$  that can capture the actual count of term  $t$  rather than the presence or absence of a term and defines as  $tf(d, t)$ . However, common terms will receive a high score using TF, which cannot capture the representative terms. To solve this

issue, Sparck Jones [1972] proposed IDF to measure a term that does not appear in many documents. IDF computes terms specificity among relevant versus irrelevant documents by normalizing the common term and rewarding representative terms, as follows:

$$idf(t) = \log\left(\frac{M}{df(t)}\right)$$

where  $M$  is the number of the documents in the collection and  $df(t)$  compute the number of the documents covering  $t$  nor the recall system effectiveness.

Typically, the term that exists in many documents over the collection serves little discriminative value in the retrieval process. For example, there are common terms (e.g., “the”) that exist in almost every document and affect the retrieval process. A highly informative term is a term that occurs in few documents in the collection. TF.IDF is a common measure to estimate both term appearance and importance. It produces a combined weight for a term  $t$  in each document  $d$  as follows:

$$tf.idf(t) = tf(t, d) \times idf(t)$$

where  $tf(t, d)$  is the frequency of the term in the document and  $idf(t)$  is the number of documents in the collection that include term  $t$ .

Relevance rankings of each document  $d_i$  in collection against a set of terms that represent as a query  $q$  can be estimated by comparing the deviation of angles between each document vector and the original query vector. After transforming the terms in a query and the documents into vectors representations, relevance is measured using vector similarity. Many vector similarity techniques have been proposed to score the similarity between documents and query vectors in the VSM. Cosine similarity measures the cosine of the angle between vectors. Empirical evidence has recommended

cosine similarity [Croft et al., 2015]. The cosine similarity is defined as follows:

$$Sim(d_i, q) = \cos \theta_{d_i, q} = \frac{d_i \cdot q}{\|d_i\| \|q\|} = \frac{\sum_{j=1}^n d_{i,j} \times q_j}{\sqrt{\sum_{j=1}^n d_{i,j}^2} \times \sqrt{\sum_{j=1}^n q_j^2}}$$

### 2.1.1.3 Probabilistic Models

Probability theory measures the possible frequency of uncertain outcomes for events occurring. When the information need is uncertain, involving probability in an IR relevance model can provide a rich mathematical framework. The probabilistic retrieval model is based on the probability ranking principle [Robertson, 1977], which states the retrieved documents should rank based on their likelihood of relevance. In other words, a given document  $d$  is relevant to a query  $Q$ , (i.e.,  $P(R = 1|d, Q)$ ) where  $R \in 0, 1$  denotes as a relevance that is a binary random variable.

In probabilistic models, documents can be categorised as relevant or irrelevant for a given query. Document likelihood is based on Bayes' theorem (i.e.,  $P(R|D) = P(D|R)P(R)/P(D)$ ) whereas query terms from relevant and non-relevant documents. Various models, such as Okapi BM25 [Robertson et al., 1994] or query likelihood [Ponte and Croft, 1998] or PL2 [Amati and Van Rijsbergen, 2002] can compute the document likelihood as retrieval function. Among all retrieval models, BM25 is probably the most popular in IR research [Zhai and Massung, 2016].

### 2.1.1.4 Language Models

Regarding language models, the simplest of these is the unigram language model that models a word at a time (also known as unigrams) where words in a language are considered through a probability. Basically, language models focus on the probability

of taking into account any sets of words that can be present in a collection, query, or document. In IR, a topic present in a query or document can be expressed through a language model. While a word in a document is not seen, smoothing strategies can be used, such as Dirichlet, Jelinek-Mercer. Smoothing techniques estimate a non-zero likelihood probability for the word that could occur in a document collection. When it comes to analysing the joint probability of terms, this method circumvents the issue of zero probability. It facilitates queries' partial corresponding wherein a document not all terms are present.

It is in one of three ways that language model based retrieval can be developed: (1) the query language model facilitating the probability to generate a document (document likelihood approach), (2) a document language model promoting the probability to generate the query (query likelihood approach), or (3) document and query distribution comparison or language models for relevance (relevance model) [Croft et al., 2015, Zhai and Massung, 2016]. Here, the relevance model is discussed at length as the experiment later in the thesis involves this approach.

The retrieval approach under the relevance model developed by [Lavrenko and Croft, 2001] evaluates the language model anticipated to be located in related documents. It uses the Kullback-Leibler Divergence measure (in short, KL divergence) to calculate the proximity between relevance model distribution and the documents. The documents that have a language model identical to that of the relevance model are considered more relevant as there is a higher representation of the related topic. Below, the KL divergence model is defined between two probability distributions,  $Q$  and  $P$ , as follows:

$$KL(P||Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$$

In fact, the initial issue for this strategy is related to obtaining the relevance model

in connection to the probability of words that appear in the relevant documents, that is,  $P(w|R)$ . When the training data is absent (i.e., relevant judgments or relevant feedbacks), the approximation of the relevance model can be made through heuristics with available evidence offered in the user information needs. Instead, a sample of assumed relevant documents can be obtained through pseudo-relevant feedback (PRF) from an early retrieval employing a standardized query likelihood model. Often, the top- $k$  ( $k$  usually ranging from 5 to 100), assumes that the documents retrieved are relevant or minimally, indicating strong relevance for a user's query. The documents serve as the training evidence needed to calculate  $P(w|R)$  directly. A relevant ranking of documents can be generated through KL divergence employing the sampled relevance model with the equivalent formula as follows:

$$\sum_x P(w|R) \log P(w|D)$$

Being highly susceptible to reliable retrieval, PRF can be quite complex [Carpineto and Romano, 2012]. Topic drift may result from noisy feedback (given the initial retrieval is inadequate) posing further issues with the process of retrieval. PRF techniques will be discussed in Section 2.1.3.

### 2.1.2 Temporal Information Retrieval

In recent years, much focus has been directed to research in Temporal Information Retrieval (TIR). This is largely due to IR's applications, tasks developing in nature and an explosion in data. The applications and challenges for TIR involve dealing with stream collections [Kanhabua et al., 2015], temporal based ranking [Dong et al., 2010, Li and Croft, 2003, Metzler et al., 2009], awareness of the query intents' temporariness [Jones and Diaz, 2007, Kulkarni et al., 2011], events revolutionary detection [Ge et al.,

2016, Zhang et al., 2015], and summarisation of time-aware topics [Kedzie et al., 2015, McCreadie et al., 2014].

Belkin and Croft [1992] undertake one of the earliest studies of the role of time importance in discovering relevant information. It is imperative when comparing information retrieval and filtering problems, and there is little to distinguish between these. It is highlighted that the time-bound status of information retrieved is a significant area to consider; none the less, it is the context that may govern temporal aspects. It is accepted that screening of information is important for different information needs. In information retrieval, Li and Croft [2003] explore the links between relevance and time in their proposed model. They call this the recency model. The authors found a major chunk of search queries prefer more recent documents. Thus, they addressed this issue by blending time with the regular query likelihood model to even both relevance and recency. Over time, the scale of document collection has been explored. This thesis has utilised the TREC microblog collection to evaluate the proposed models and compare with the recency model.

Kleinberg [2016] reviews the current strategies and technologies in the temporal areas of information. The author states that data can be identified as an information stream instead of a static dataset. Examples of these sets could include news, patents, scientific articles and emails. There are two categories that can be used to define these information streams: (1) bursty (i.e., occasional), where there is reduction or growth in the topics over time as well as in their intensity, and (2) the topics/subtopics may be the focus of momentarily co-located documents but the stream is merged for the user. These complexities must be addressed by the system that filters information from such a stream.

### 2.1.3 Query Expansion

Query Expansion (QE) is the process of boosting a given user's query with extra features (e.g., words or terms) that are closely related in order to augment the IR system performance. The IR system is improved by considering that, to meet user's information needs, users normally coin different descriptions for their query. Specifically, QE's aim is to enhance the capability of documents matching and topic coverage triggered by a query and, hence, augment the relevance ranking model. In relation to a search engine, expanding a query focuses on assessing a user's input (the words used to make a query and other data (at times)) and escalating the query to retrieve a larger number of documents.

Over many years, the focus of research has narrowed to QE as the approaches has proven adept at addressing the problem of vocabulary mismatch. It is the difference in the textual representation of queries and documents that gives rise to the real problem. For a given query, the set of terms entered by the user may not be present in the relevant documents, which may result in relevant documents not being retrieved. Carpineto and Romano [2012], conducted holistic research was on QE methods offering observations on the problems facing the QE approach. Their work highlights critical difficulties including the parameter settings, efficiency, and usability of approaches. This area has made an impressive contribution to the design and augmentation of a query that relies on Pseudo relevance feedback (PRF).

#### 2.1.3.1 Pseudo Relevance Feedback (PRF)

Pseudo relevance feedback (PRF), also known as blind relevance feedback or local analysis, offers a technique for Automatic Query Expansion (AQE) via local analysis. It automates the manual part of relevance feedback methodology without any efforts

from the user with annotated the relevance documents. PRF is employed in automatic query expansion techniques that, providing an initial query, presumes relevancy for the top documents in an initial retrieval. The feedback is deemed as pseudo-relevant owing to the absence of relevant knowledge. From the top pseudo-documents, terms are selected and then ranked based on a weighting function to enhance the original user query. As a result, PRF provides a reliable and lightweight means to locate and rank the best possible expansion evidence for an initial query. Effectiveness is tried to the collecting of a good pseudo-documents, as relying on PRF alone does not produce a perfect result. In explicit relevance feedback, the first retrieval result phase is visible to the user, which requires judging of the document in the list that is relevant or not to their need.

PRF is a central concept in AQE [Xu and Croft, 1996] and has been proved in the IR research [Lavrenko and Croft, 2001, Zhai and Lafferty, 2001a]. In a comparative analysis conducted by [Zhai and Lafferty, 2001b], one of the most successful pseudo-relevance feedback approaches was found to be RM3, a version of the relevance model. Since there is no guarantee that the top documents will always be relevant, the result may be unstable or be impacted by a topical drift. Topical drift occurs when the concentrate of the search topic moves to an inadvertent topic affected by improper expansion features [Zhai and Massung, 2016]. PRF's success in multiple retrieval scenarios shows that valuable information can be obtained from the top documents result [Lavrenko and Croft, 2001, Lv and Zhai, 2010, Qian and Zhai, 2013]. In this thesis, we revisit the PRF assumption in the proposed models for improving the effectiveness social media search.



## 2.2 Information Analysis in Social Media

### 2.2.1 Overview of Social Media

Social media is a fascinating idea; it is an interactive medium to bring people together. McCay-Peet and Quan-Haase [2017] offer a definition of social media as “web-based services that allow individuals, communities, and organisations to collaborate, connect, interact, and build community by enabling them to create, co-create, modifies, share, and engage with user-generated content that is easily accessible”. There are three major features that we can dissect from this definition. The first feature is to produce and share content. This points to the spread of user-generated content (UGC) and user-published data. The second component is the virtuality of social media, as indicated by applications and websites. Finally, the aspect of connectivity among users demonstrates a social networking ability.

There are at least eight categories in which social media platforms and technologies can be grouped: e-commerce gateways; microblogs (e.g., Tumblr, Instagram, Twitter); social networking (e.g., LinkedIn, Facebook, MySpace); multimedia portals (e.g., Vimeo, Twitter, Facebook, Periscope, TikTok, YouTube); virtual worlds (e.g., Second Life); review platforms (e.g., Tripadvisor, Foursquare); and social gaming (e.g., World of Warcraft).

When someone uses social media, they acquire a form of new media in themselves. In addition to absorbing/consuming information, users also propagate/produce it in relation to the world around them (such as celebrity gossip). UGC includes different types of content [Baeza-Yates et al., 2011] such as the interactions that the users undertake with already published content [Volkovich and Kaltenbrunner, 2011]. UGC is defined by Baeza-Yates et al. [2011] as: “one of the main current trends in the Web.

This trend has allowed all people that can access the Internet to publish content in different media, such as text (e.g., blogs), photos or video.”.

While social media content is valuable, UGC provides significant challenges. First, users of social media often produce real-time text that does not necessarily represent meaningful content. Second, users often do not follow the proper English structure when publishing on social media, leading to meaningless content. Third, due to its restriction, users are unable to provide much text on most social media platforms. The lack of content is the leading cause of the sparseness problem; not enough statistical information can be extracted from short text content. Raw content occurs at an unprecedented rate, meaning content on social media can be overwhelming. In the case of Twitter, more than 500 million tweets are sent each day by more than 100 million active users<sup>3</sup>. Finding relevant information can, therefore, be time-consuming.

Different types of microblogging technologies are available within social media to help achieve a number of aims. Twitter is a microblogging service introduced in March 2006. With over 125 million daily active users, Twitter is ranked among the most popular social media platforms. The founders of the platform defined Twitter as an unimpeded opportunity that allows everyone to create and share information and ideas in real-time. “Tweet” is a term that refers to a short text message that a Twitter user can produce (e.g., shown in Figure 2.2.1). This short plain text (tweet) can also include videos, photographs, and website URLs. Until recently, Twitter allowed 140 characters for a plain text message; however, in November 2017, the length was expanded to 280 characters. The service is based on the followership principle whereby users may follow others and be followed. A user can follow anyone on Twitter (i.e., as long as their account is not private). It is not necessary that the one followed needs to follow back. The “@username” feature can be used to mention a specific person for

---

<sup>3</sup><https://www.fastcompany.com/90256723/twitters-q3-earnings-by-numbers>



**Figure 2.3:** An Example of Tweet.

interaction. There are two different ways by which a user can interact with a published tweet: appreciation of the tweet may be expressed by pressing the like button (and thereby saving it as a favourite tweet) or by retweeting (i.e., forwarding the tweet to your own followers). Moreover, the hashtag feature can also be used to annotate user messages where the prefix “#” character is used as a non-spacing word. The user can use the hashtag to emphasise the major topic in a tweet. The hashtag also helps refine a search. Users may use a hashtag to find relevant information. All the tweets using a given hashtag are easily available under that hashtag. In this thesis, we focus on Twitter content to discover relevant information.

### 2.2.2 Microblog Retrieval Models

Retrieving information should take into account the nature of social media, including specific characterisations and social networks. In regards to these two aspects social networking and content generation several research areas surface simultaneously, including, critically, social connectivity analysis. The structural variance in microblog documents makes retrieving microblogging distinct from other retrieval tasks. Content

generated by a social media user is specific to them. This specificity has given birth to new tasks of information retrieval that match the user's information needs. This thesis focuses on meeting the user's information needs by locating relevant information from microblog streams. This section reviews the existing literature pertaining to microblog retrieval.

Microblog users who search over microblogs can be considered in two categories based on their information needs [Efron, 2011]. In the first category, users look for answers to questions they ask expecting the answer from their network and circle of friends. Here, the interaction occurring around the information search is almost identical to that of mailing lists, Q&A platforms (such as Quora) or other discussion-based portals where questions asked by a user are answered by others on the forum. In the second type, users carry out searches in the microblog similarly to an ad-hoc IR, which is the centre of discussion in this section.

### **2.2.2.1 Social features based models**

Similar to the commonplace web search, users can publish their tweets, use hashtags within their tweets, keep updated about another user's tweets by following them, or use their needs-based keywords to make a search query. Mentions, hashtags, and URLs are included in microblog metadata. The context and content of the tweet is defined by this metadata. Though some authors have considered it a relevance indicator to highlight user produced data [Duan et al., 2010, Tao et al., 2013], others favour use of hashtags to exploit this data for tweet and query enhancement [Efron, 2010a] as well as URL-attached content Jin et al. [2011], McCreadie and Macdonald [2013].

**Hashtag:** Efron [2010a] developed a method that returns a ranked list of hashtags that are ranked based on their relevance to a given user query. They argue that hashtag-based searches can offer many advantages, including enabling users to discover relevant/interesting hashtags to derive information or follow them. Such a search result can also be helpful in grouping results under closely related hashtags, and the ranked hashtags may be useful to expand queries. For every unique hashtag, a language model is trained in the data employing the tweets. These tweets have a specific hashtag that is later ranked with relevant hashtags. Evaluation of the results was done using a tweet corpus with 29 queries and corresponding relevance judgments that were labeled by human experts. In another study by [Lin et al., 2011], ten hashtags were manually chosen based on their reputation as topics. For each topic, they developed a specific language model. Then, the language models were exploited in order to calculate scores of relevance on tweets appearing in the tweet stream and to eliminate any irrelevant tweets. On topic-based language models, the examination was done for four different smoothing techniques supported by background models to address the problem of sparsity probability. All of the topics were steady, organised, and (over a passage of time) non-advance implying that in the search, the issue of topic drift was skipped.

To expand tweets, URLs and hashtags were employed by [Sharifi et al., 2010]. Expansion of a tweet is done by attaching a precise representation of the inherent hashtag. Terms that appeared most frequently alongside a hashtag were taken to be discriminative for that particular hashtag. Using the same method, tweets containing URLs were also expanded. For a given query, a two-step process of ranking is followed. Step one uses recency data and similarity of content in ranking. The second ranking draws only on tweets that ranked in the top of the first ranking. This second ranking weights tweets based on components like URL popularity, tweet impression, user popularity, and a score of authority by users. In their experiment to assess the

TREC microblog dataset 2011 results, the authors noted that it was not so advantageous to expand tweets using hashtags as they developed to an extent that may reduce the search effectiveness in some cases. The use of hashtags was also investigated by [Laniado and Mika, 2010]. The study offered techniques to identify hashtags that may be devoted to real-life events as well as to users. [Duan et al., 2012b] presents a method to classify tweets in a graph optimisation framework that presents information in six umbrella topics: lifestyle, politics, sports, entertainment, business, and science and technology. Associated tweets that have the same URL or hashtags were used to optimize every tweet's representation and improve the model in a trainable fashion. Hashtags were used as a substitute for user feedback.

**User Network:** Two ranking methods were developed by [Nagmoti et al., 2010] to rank tweet's authors. These same measures were then used to rank tweets in association with other features particular to the tweet. Tweet authors are then assigned a score under the author ranking method. The first ranking method measures the total number of tweets by a user/author. The second method considers the total number of followers/followed based on the notion that users will naturally follow an author who produces meaningful content. Thus, based on the author scores, tweets may be simply ranked again in the first retrieved result. The authors also developed two additional features of this formula to re-rank tweets: tweet length and URL presence in a tweet. They evaluated their model with a labeled collection, and the experiment results showed the second measure, built on the count of follower/followed, was more efficient compared to the first, which was based on total published tweets by a user. It was also shown that tweet length is a better criterion of substance in a tweet, though tweets are still short in general.

A combination of a tweet's trustworthiness and relevance is assessed by RAPPop

[Ravikumar et al., 2013]. Firstly, a tweets is assigned a feature score that weighs the trustworthiness of the tweet's source. The graph includes users, tweets, and URLs that the tweet refers to. Calculation of trustworthiness is carried out with consideration to total tweets, duration of the profile, number of followers/followees, and relevant profile information. A tweet's trustworthiness is calculated with total tweets, hashtags used, and other features. The PageRank score is used to determine a URL's trustworthiness. Agreement analysis, the second method, calculates a tweet's content for its trustworthiness. The authors assessed RAPRop employing the TREC microblog 2011 dataset. The P@30 metric was optimized by RAPRop, producing an improved result when compared to Twitter's current search function.

**Geo:** Several research directions have been proposed in regards to geographic information evidence. [Hong et al., 2012] presumed that every tweet was produced from three kinds of language models: the topical language model, the language model pre-region, and a background language model. First, for every tweet, the selection was made for the latent location and the region of a tweet. Later, concerning the region and the user, the topic was chosen. In this experiment, observations were made about different patterns in the topics and regions. [Kotov et al., 2015, 2013] used the Latent Dirichlet Allocation (LDA) and the Latent Variable Model (LVM) modelling method to expand a geography-based extension to produce topics that were sensitive to geographical topics. Their framework was based on language model retrieval.

**URLs:** Damak et al. [2013] argue that mentions, hashtag based features, URL specifics and term-based features are more effective when compared to replies. Similarly, the Twitter TinyURL method (i.e., shortened links to full URL) was proposed by Chang et al. [2013] to identify documents of superior quality and relevance and to influence

data on Twitter to produce high-quality and efficient features to be used for document ranking. Other authors highlight the importance of the content of URLs linked in a tweet [McCreadie and Macdonald, 2013]. Likewise, [Luo et al., 2012] suggested employing a ranking method taking into account features from a tweet's metadata like URL presence, frequency of retweets, and hashtags, replies. The authors offered that we can improve performance to a great extent by including these additional features. Evidence on microblog searching suggests that there are many features other than the similarity of content on which a tweet's relevance may depend (e.g., freshness level of information, the location of the user, URL presence, and a number of followers/followees) [Duan et al., 2010, Nagmoti et al., 2010].

### **2.2.2.2 Time aware Features Models**

The user usually expresses their need explicitly using a query. During their interactions, other indirect and implicit information needs may also arise, such as time and situation. Several aspects of microblog retrieval are governed by the importance of time, which can help discern different types of information as new needs emerge with the evolution of current topics. This section reviews the latest studies for time-based microblog search.

Two major directions can be used to categorise the current research on time-aware microblogs ranking: (1) ranking based on recency and (2) ranking based on time. The first ranking often optimizes microblog and social media ranking [Kanhabua et al., 2015]. There is an implied real-time information need when examining search queries made by users on Twitter [Teevan et al., 2011]. Thus, in social media searches, the recency-based ranking can be effective.

Yet, according to Jones and Diaz [2007], recency-favouring queries are merely a



subset of the temporal queries in which the effect of the time-based features of queries in relation to the effectiveness of tweets retrieval can be divided into two types: (1) sensitive to time and (2) insensitive to time. The authors also found the need to proximate relevant time intervals for time-sensitive queries and to combine this time-based relevance into the model of ranking, which distributes datasets into temporal and language-based evidence. Choi and Croft [2012] proposed a model that manipulated the time-based distribution of a user's retweet trend to allow for query expansion over a period of time. It is noted in the TREC experiment datasets that retrieval performance is optimised through the temporal relevance model. The temporal cluster hypothesis is behind this approach. The hypothesis offers that identical temporal features are shared by relevant documents.

Efron [2010b] examined the weight of the term in relation to its time-bound function to a certain time point. They assigned weights relevant to how effectively, over time, the frequency distribution of a term works under a linear model. The authors contend that, compared to more common terms, more frequent change behavior is noted when terms are more discriminatory and show more uncertain behavior. Efron et al. [2014] offered ways to assess the temporal density of a relevant dataset and, to derive a benefit for tweet search, explored the role of temporal feedback. A ranked list of related documents was generated by a language model that was based on the query likelihood model. A log-linear model integrated the relevance probability, given temporal features, with the relevance of term-based probability. Chen et al. [2018] deployed a word-level temporal predictive method to expand temporal feedback, at the document level, so that more fine-grained information can be obtained. They combined this evidence for time-sensitive ranking in microblogs by optimising the information at the word-level temporal relevance. It was also noted that PRF benefited from this temporal relevance. When incorporating the word temporal relevance estimation

into PRF framework, the performance result of the final retrieval was significantly improved using only document-level temporal feedback.

### 2.2.2.3 Learning to Rank based Models

A family of machine learning methods, L2R (learning to rank), is used to learn a ranking function under machine learning environment. This technique commonly represents a query-document as features vectors, with the availability of relevance judgments, to learn a ranking function. Based on the training data that learns with ranking function, it is then used with testing documents. The flexibility of the L2R method is its primary strength as it can combine several types of evidence into the retrieval process [Liu, 2009]. This section reviews the learning to rank models within the microblog context.

Duan et al. [2010] develop a method to rank tweets by evaluating the tweet's content features, its user's authority, and features related to that specific tweet. The authors focused on three features related to content: BM25, tweet-length and content similarity. In BM25, they used TF-IDF measurement to estimate the relevance of a query-tweet. For content similarity measured, they inferred tweet popularity in the collection. The tweet length is measured through the total number of words used. PageRank algorithm offers the user account authority. Fixed Twitter-specific features were used. These are URL count, URL presence, hashtag count, count of retweets, a reply or main tweet, and vocabulary-based word ratio. Experiments were done with microblog collection using twenty queries in a topic set that was labeled by humans. Experiment results were evaluated on a five cross-validation through the RankSVM algorithm [Joachims, 2006], which had properties as noted above. They concluded that, when ranking a tweet, tweet length and URL presence information are more important for consideration. An identical approach was offered by [Cheng et al.,

2012] that also combined temporal features. Along with RankSVM, the authors used LambdaMART [Wu et al., 2010], which is an approach to learn listwise ranking.

In a microblog search environment, Wang et al. [2014] employed L2R strategies to accumulate the relevance features. They categorised the features as two types: entity-related features and temporal-related features. Temporal-related features offered some insights to the document's temporal distribution, the average time of the whole dataset, and the time distance between a document and queries. Next, all of these features are combined using the L2R algorithm [Cao et al., 2007]. Combining temporal evidence into ranking models was shown by the authors to be an efficient approach.

Qiang et al. [2013] used Factorization Machines by combining the pairwise L2R method for efficient retrieval within a microblog. For Factorization Machines, two directions were used to enhance loss function: Adaptive Regularization and Stochastic Gradient Descent built on the work of [Rendle, 2012]. The improvement was noted in the model in this experiment; however, the differences were not very clear at the baselines. As L2R needs well-defined steps for learning, [Berendsen et al., 2013] developed pseudo test collections to learn L2R concerning tweets linked to a hashtag. The authors employed four schemas to choose a hashtag in focus: all hashtags, random order, timestamp dependent on the relevance judgment. The experiment fell short of the expectation as it didn't show a robust relationship while choosing sets for training.

#### 2.2.2.4 Fusion based Models

IR Fusion refers to the production of a single result by joining various sources of information in response to a single query. IR Fusion can be achieved by joining the results of several algorithms that rank datasets, representations from various documents and user information needs, or a mixture of all of these. Microblog retrieval

research has invested sufficient energies in the data fusion methods [Liang and de Rijke, 2015, Liang et al., 2013, Losada et al., 2018]. As a result, quite a few ranking algorithms have been developed. Common examples include CombSUM [Shaw and Fox, 1994],  $\lambda$ -Merge [Sheldon et al., 2011], supervised rank aggregation [Liu et al., 2007], Borda data-fusion [Aslam and Montague, 2001], fusion for divergence [Liang et al., 2014a], time-sensitive accumulation [Liang and de Rijke, 2015, Liang et al., 2014b], cluster-based fusion [Khudyak Kozorovitsky and Kurland, 2011], rule-based aggregation [Caragiannis et al., 2019], and accumulation algorithm, which acquires linked models on object features as well as on lists [Bhowmik and Ghosh, 2017].

Liang et al. [2014a] and Liang and de Rijke [2015] identify about data fusion/rank aggregation as a significant method for retrieving information. This technique joins different lists of ranked documents that have been retrieved from a corpus against a query through several algorithms at the retrieval process. Any approach to retrieval can generate these ranked lists by undertaking various actions and representations of documents and/or queries. It is hypothesised that when combined together, several retrieval approaches optimise the process to produce the ultimate fused list.

To date, data fusion research has involved the independence of documents in merged lists, and only documents that are high on many lists have been relevant in a given query [Liang et al., 2014a]. It is stated in the cluster hypothesis that in the same internal structure (i.e., multiple or cluster) documents are likely to exhibit identical relevance features to the same query attempting information retrieval [Liang et al., 2014a]. In data mining and information retrieval, this concept has been successfully tested. Khudyak Kozorovitsky and Kurland [2011] inform that it is merely to a limited degree that cluster hypothesis has been applied in data fusion and results have demonstrated poor efficiency. A burst-sensitive method was developed by [Liang and de Rijke, 2015] to fuse lists of documents retrieved against a query by combining

information utilized by fusion techniques curbing time-aware document clusters.

### 2.2.3 TREC Microblog Retrieval Tracks

The “Text REtrieval Conference” (TREC<sup>footnote</sup><http://www.nist.trec.gov/>) is an international conference that is devoted to the improvement of information retrieval researches in a number of fields. TREC has continued for more than twenty years with the aim of improving networking among industry professionals and academics, as well as promoting large-scale system evaluations. TREC is supported by the U.S. Department of Defence and National Institute of Technology (NIST). Since 2011, TREC has organised a “Microblog track” that includes a main real-time ad-hoc task and a second filtering track (introduced in 2012). The main aims of this track were to address the challenges of microblog retrieval through design innovation and the evaluation of microblog search systems. The first dataset was collected in 2011 but was utilised for tasks during both the 2011 and 2012 tracks. The dataset, called Tweets2011, was gathered over 16 days (from January 17th to February 2nd, 2011) and includes around 16 million tweets using Twitter Stream API.

In the same way, a second dataset, Tweets2013, was collected in 2013 and utilised for both the 2013 and 2014 microblog tracks. This second dataset covered two months of Twitter stream and includes approximately 260 million tweets. TREC organisers produced two test sets for each dataset and provided a relevance judgment to evaluate the performance of the submitted systems’ runs. The total number of queries for each test was between 50 and 60. In this thesis, we used both datasets and all test sets. More details about the datasets are covered in the experiments chapter.

In the same way, a second dataset, Tweets2013, was collected in 2013 and utilised for both the 2013 and 2014 microblog tracks. This second dataset covered two months

of Twitter stream and includes approximately 260 million tweets. TREC organisers produced two test sets for each dataset and provided a relevance judgment to evaluate the performance of the submitted systems' runs. The total number of queries for each test was between 50 and 60. In this thesis, we used both datasets and all test sets. More details about the datasets are covered in the experiments chapter.

In 2011, TREC started the microblog search track. Here, systems are asked to return relevant documents for a given query at a specific time [Ounis et al., 2011]. Many approaches have been proposed with consideration of microblog characteristics (such, embed URLs and hashtags) but using different techniques. These techniques were included query expansion rank [Amati et al., 2011], learning to rank [Metzler et al., 2011] that achieved the best P@30 performance overall runs and query expansion from an external resource, such as Google API [Bandyopadhyay et al., 2012, Louvan et al., 2011].

A filtering task was first run at TREC 2012 as the reverse task to the ad-hoc search task. The user information need was defined as a query and a specific time [Soboroff et al., 2012]. [Han et al., 2012] used a KL divergence retrieval model to compute the estimated difference between a document and a query model, taking advantage of embedding URLs in the top retrieved tweets. Zhu et al. [2012] also used query expansion that combined Google search API results with learning-to-rank algorithms.

## **2.2.4 Microblog Applications**

### **2.2.4.1 Microblogs Summarisation**

Summarising text is a daunting activity that derives implicit information from a document and secures the embedded meaning [Nenkova et al., 2011]. Summarisation is

achieved through two major strategies: (1) extraction and (2) abstraction. Whereas extraction focuses on sentence-level information, abstraction produces sentences/phrases not present in the actual documents. It is suggested that both of these methods can solve the issue of microblog summarisation. Existing literature notes that most methods devote attention to selecting tweets with important statuses to represent a given topic. Summary generation of microblogs focuses on producing a digest for not only long-active but also completed incidents from streams of tweets to acquire a picture of events around a topic or a user's perception of a topic. As of late, TREC has introduced microblog summarisation in its tracks.

A dual wing-factor graph model was developed by Yang et al. [2011] to combine tweets into the process of summarisation. To select tweets and summary sentences, the authors defined a selection criterion. Wei and Gao [2014] used cross and local features (used for headline mining) to develop an L2R model. Further, Wei and Gao [2015] used relevant tweets to extend LexRank [Erkan and Radev, 2004] to develop an HGRW (heterogeneous graph random walk) for single document summarisation. To summarise individual documents, Nguyen et al. [2018] blended user posts with related documents. The results produced by the model were promising; the results covered two different languages and three databases. Nguyen and Nguyen [2017] scored tweets and sentences by exploring the sentence-tweet link through a set of lexical level properties. A reinforcement process is achieved to calculate the score of tweets and sentences. Through this process of extraction, the models offered the highest-scoring tweets and sentences; however, issues arose around hand-annotated tweet-sentence features, domain specifics, and the complexity of showing user post and sentence relationships.

For extractive summarisation, it takes shortlisting a related subset of tweets having redundancy as less than practical to capture the major aspects in an event. Two

approaches, graph-based and feature-based, were developed within extractive summarisation to discern a tweet's relevance. In the first strategy, a graph models a tweet stream. Here, a vertex represents a tweet and the similarity between the tweets are connoted by an edge [Duan et al., 2012a]. Relative to the term frequency and bursty, tweets in [Duan et al., 2012a] are grouped, and each group is ranked as per a salience score it receives. Relative to the total followers and tweets made, Liu et al. [2012b] clustered the content of tweets for similarity, along with social similarity between users, to calculate the weight of an edge. Vertices with the highest salience score are used to build the summary.

From the traditional approach of summarising documents, Inouye and Kalita [2011] examined two graphical algorithms to summarise tweets. These algorithms were LexRank [Erkan and Radev, 2004] and TextRank [Mihalcea and Tarau, 2004]. Besides the statistical features of the text, these methods manipulate relation among tweets. In the first method (LexRank), the similarity of two tweets is represented by the edge weight, and a tweet's final score is calculated relative to the weight of connected edges. The other method, TextRank, works with the algorithm of PageRank and combines the graph's whole complexity as apposed to merely grouping similar pairs (e.g., as in LexRank). A tweet's ultimate score is calculated recursively dependent on the weight of the edges that are directly related, including edge weights of additional tweets related to the tweet at hand. The generated outcome shows that, when compared to graph-based approaches, feature-based summarisation methods work better. It also suggests that owing to the interlinked complexity in LexRank and TextRank, summarisation of tweets did not benefit much from the algorithms.

For summarisation of tweets, current research suggests using a social-temporal scenario [He et al., 2017]. The proposed strategy relies on LexRank. Here the edge weight is calculated by grouping similar tweet content with the social context of the



author and the temporal context of the tweet itself. The user's authority defines both the social context in a social network and the tweet's popularity (i.e., total retweets). The update rate of tweets defines the temporal context of a given topic. In summary, the Maximal Marginal Relevance (MMR) algorithm is used to circumvent redundancy. In this algorithm, as soon as a tweet is included in the summary, the re-ranking of other tweets takes place as per the dissimilarity of the summary of the tweet. Cosine similarity is then used to calculate the similarity between two tweets.

There is another extractive method that works to the features. Features based methods mainly rely on statistics of text in tweets, including language model [Fan et al., 2016], term frequency [Liu et al., 2011], TF-IDF [Chakrabarti and Punera, 2011], Temporal TF-IDF [Chong and Chua, 2013], and Hybrid TF-IDF [Sharifi et al., 2010]. Two common techniques have been utilised in previous studies to select tweets for incorporation in summary. Firstly, the summary is formed using the top  $m$  tweets. Here,  $m$  signifies the summary's wanted limit length. Sharifi et al. [2010] developed a hybrid TF-IDF method. In the approach, the TF features are computed over the tweets and taken as one document. Top tweets are iteratively extracted by excluding the tweets that have cosine similarity over the predefined threshold in the light of evidence relative to the summary of tweets. One of the early summarisation methods was developed in [Lipizzi et al., 2016]. This method watches the live tweet stream as it happens. Timed events (e.g., a show or sports match) are covered by this approach, which relies on term frequency to calculate tweet relevance in regards to a current event along with KL deviation to minimise redundancy. Sumblr, developed by Shou et al. [2013], is an incessant summarisation method based on clusters. It offers both online and historical summaries. Tweet inclusion, in summary, is done by clustering tweets and selecting those with the greatest scores in every group.

In the next category, summary generation is produced as a problem of optimisation.

In this scenario, ILP (Integer Linear Programming) grouped with clustering methods has been employed in summarising several documents [Li et al., 2011, McDonald, 2007]. To summarise a microblog, Liu et al. [2011] suggested a concept relevant ILP strategy. This method first mines, for every topic, a group of  $n$ -grams that are found in tweets occasionally concerning a topic but not in a corpus. The mined  $n$ -grams are taken as concepts. A group of tweets is chosen to construct the summary. These tweets offer as many prominent concepts as practicable along with the objective function to augment the total of the weight of concepts and constraints with the length (e.g., words and tweets) and the coverage of concepts.

#### **2.2.4.2 Microblog Personalisation**

Users consume and create a variety of items in their online lives, such as bookmarks, current context, and search history. These items are used in content-based personalised methods to develop the user's representation; those representations are then used to adapt search returns with attention to the user's search needs. To achieve a personalised search, a number of content-based approaches have been suggested to satisfy a user's information needs. The strategies that utilise social media data as information sources are discussed in this section.

Though search, as specific to microblogs, has been explored to some degree, personalised microblog search remains a highly unexplored area. Some work has been done in relation to re-ranking personalised tweets [Feng and Wang, 2013, Li et al., 2016, Zhao et al., 2016b]. In a user's timeline, Feng and Wang [2013] re-ranked tweets relative to the possibility of these tweets being shared. Zhao et al. [2016b] infers a user's interests through the WeMedia accounts a specific user follows. This information is then applied to re-rank their tweets.

Making use of filters, Zhu et al. [2017] suggested a real-time approach to personalised search. The profile of a user is explained as a group of boolean operators; only those tweets that meet these boolean rules are selected, while all other tweets are directly unselected. The similarity is then calculated, and a score awarded, based on the chosen tweets and the query. These tweets move past the profile of the user to be re-ranked. With an external search engine, they also expand the tweets by exploring the keywords a tweet contains. Results are then evaluated relative to a tweet sample acquired from the Twitter API; however, the paper does not discuss the particular detail of selecting queries and grounding accurate results against a user's query. Today, many approaches exist that make use of a user's social networking graph for personalised results.

A time-sensitive user behaviour model, TPM (Tweet Propagation Model), was suggested by [Ren et al., 2013]. The model follows topics and interests dynamically related to the user. Three different categories are used to divide the topics: bursty, common, and personal. After a tweet's probabilities are inferred, top-ranked tweets are selected through an iterative process to augment novelty, coverage, and the summary's diversity. A common ranking method was used by [Chen et al., 2012] to recommend tweets. It uses many features related to the tweet to impact its importance. Current effort does not focus on the user's social connection and topic diversity when developing the user framework. A more developed user model is suggested by [Abel et al., 2011] to optimise the process of recommending news to a user. They explore several means of modeling a user profile, employing entities, hashtags, and topics. The study concludes that results are best under this entity-based modeling. A simple algorithm recommender is used by the authors employing cosine similarity between tweets and user accounts.

In recent years, different studies have found query expansion to be advantageous

for retrieving microblogs [Bouadjenek et al., 2013a, 2011, 2016, 2013b, Zhou et al., 2012]. Based on the topic model, Zhao et al. [2016a] suggested a personalised hashtag ranking that can mine covert topics. This hashtag-LDA model and employs an experimental approach using actual Twitter data. The authors note that the hashtag-LDA model offers personalised hashtag results per the latent topic information in untagged accounts. For latent topic mining, this model can optimise topic generation by grouping words and hashtags together. Through this strategy, the authors developed their model, using Gibbs sampling, to discover the latent topics and take into account real-time Twitter data to assess their method. More details regarding utilising topic model strategies for microblog content will be discussed in Section 2.3.1.

#### **2.2.4.3 Opinion Retrieval**

Social media has become the normalised way for the general public to take part in global debates (on both politics and social issues), as they share their opinions and express their positions. In turn, users are searching for information as a benefit of crowd-sourced opinion on social media (e.g., looking for hotel prices). Relevant posts can be found using opinion retrieval approaches to confirm either the pros or cons of a topic. In the event of large-scale campaigns or political events, reliance on social media is prevalent in both the masses and influential circles as a means to access and disseminate information. The challenges faced by opinion retrieval methods lie in linking sentiment to detected opinionated content, where there were positive, negative, or neutral emotions [O'Connor et al., 2010]. Both supervised [Popescu and Pennacchiotti, 2010] and unsupervised [Bernstein et al., 2013] strategies were employed to study opinion detection within microblogs. Fang et al. [2015] investigated trends in the voting of individuals during the referendum of Scotland in 2014: “No” opposing it and “Yes” favouring it. To classify user’s voting trends, the authors suggested a topic-based

Nave Bayesian classifier relying on tweet content. To examine the same content, Liu et al. [2012a] suggested employing a dataset labeled by a human to develop a language model that was smoothed to utilise noise in emotion data.

Topic modeling has also been applied to social media to discover viewpoint. A time-sensitive topic model is introduced by [Ren et al., 2016] to summarise contrary opinions relying on emotion (e.g., neutral, negative, or positive). The unsupervised topic model, SNVDM model (Social Network Viewpoint Discovery), is proposed by [Thonet et al., 2017] to recognise the topics and viewpoints of several users. Aside from the content produced by users for a given topic, the model also makes use of the social activity a user undertakes on social media. The model assumes that interconnected users share the same or similar viewpoints. [Meng et al., 2012] introduces an entity and topic-based opinion mining approach to mine summaries concerning opinions and topics.

#### **2.2.4.4 Topic and Event Detection**

In this section, we introduce the methods developed to aid topic and event detection in microblog streams. From the traditional web's topic detection approach, topics and events detection methods for microblog streams can be categorised into general and specific classes. These classes are categorised based on the discovery of topics relative to the topical nature, along with new topic and retrospective detection as per the task of detection and its target for applicability. Next, these techniques are discussed at length.

Concerning the topic of interest, topic detection techniques can be classified into unspecified and specified methods based on available information related to the topic. The methods for unspecified topic detection often locate the occurrence of a new topic

in microblog streams. Owing to the absence of information about the given topic, identification is based on temporal evidence. Typically, when applying these techniques to microblog streams, the first step is to locate the bursts out of which similar topics are combined and categorised into various subcategories. Contrarily, methods for specific topic detection usually rely on specific information already known for a topic (e.g., names, time, and place). Thus, given these pieces of known information, the detection process can be aided by traditional approaches to retrieve and extract information.

Unspecific often include breaking news, rising topics, and daily topics, which appeal infinitely to microblog users. While several features unique to microblogs have been used to detect unspecific topics, these methods produce results that are limited to posts exhibiting only the typical characteristics of microblogs. These topics are typically located by exploring the temporal patterns or signal in relation to the post's streams. A burst of features is usually present in a new stream of topics (i.e., a rapid increment in specific keywords). Such posts can thus be accumulated into trends [Mathioudakis and Koudas, 2010]; however, microblogs include both relevant and irrelevant evidence in trends. For example, a system was suggested by [Lee and Sumiya, 2010] to detect Twitter topics that are controversial, such as different opinion topics. Based on a framework to detect crowding activities, [Lee and Sumiya, 2010] proposed a process for detecting a geo-social local topic system. The process works through watching the behaviour of the crowd. Similarly, a typhoon and earthquake detection system was developed by [Sakaki et al., 2010] that works by monitoring posts on microblogs. Therefore, non-topic trends need to be distinguished from trending topics. Moreover, as the volume of microblog posts is massive, it is imperative to consider efficiency and scalability.

In terms of handling the specific features of the microblog, various works have been proposed to enhance the performance of detection. For instance, [Petrovi et al.,

2010] applied a clustering strategy to detect specific topics. Rather than focus on all the content in a microblog, the authors focused exclusively on noun terms, text features, hashtags, and user names to calculate post similarity. Topical words are mined to identify the frequency of terms in the hashtag and to detect emergent topics [Long et al., 2011]. To detect the presence of fresh topics, as the topical words are facilitated, a maximum weighted graph (i.e., bipartite) matching is used to form chains of topics.

While it is new topic identification that has become the focus of most attempts, some efforts were steered toward identifying retrospective topics from past microblog posts. For new topic detection, an approach was proposed that combined matrix factorisation and dictionary learning. In [Kasiviswanathan et al., 2011], a suggestion was made for a dictionary learning method. The method has two stages: discerning novel documents from the stream then discovering cluster structures that exist among subsequent documents. In a related study, [Saha and Sindhwani, 2012] employed a non-negative matrix factorisation model in conjunction with a temporal normalisation. The temporal normalisation is established by joining trend mining with a margin-focused loss function to penalise decaying or static topics. For microblog data, current search engines (e.g., Google and Twitter) are limited to returning stand-alone posts in response to a query [Metzler et al., 2012]. The attempt to locate relevant messages in relation to a given query faces two massive challenges: the dynamic evolution of vocabulary mismatching and the posts' sparseness. For example, query keywords may not exist in the associated posts or different hashtags and abbreviations may be used to represent the same topic. Conventional query expansion approaches typically use co-occurring words along with keywords. To retrieve microblog data on topics, we should use dynamic and temporal query expansion methods as proposed models in this thesis.

### 2.2.5 Query Expansion in Microblogs

Many queries are submitted through microblog platforms (such as Twitter) to obtain useful information. These search queries are typically short. Thus, many search results are often irrelevant to the query keywords. A word mismatch indicates that users regularly use several a set of words to characterise ideas in their queries that a user may use to depict the same ideas in their documents [Li et al., 2012]. The core issues in ambiguity stem from hyponymy and synonymy. Various approaches for handling word mismatches have long been considered [Albathan et al., 2013, Li et al., 2015b, 2010, Miyanishi et al., 2013]. Therefore, taking into account the query words will improve the relevance of retrieved documents.

Query expansion is a well-known methodology for managing vocabulary mismatch. The objective of query expansion is to expand an initial unsuccessful query with different words that best encapsulate the user's intent or that create a relevant query likely to retrieve more significant documents [Carpineto and Romano, 2012]. The method is especially valuable when the user's query is ambiguous, short, or needs useful words relating to the expected topic. This procedure can be automatic, through pseudo-relevance feedback, or it can be manual using explicit relevance feedback.

In response to microblog document challenges, previous works demonstrated that query expansion could improve microblog retrieval effectiveness. A web-based query expansion technique has been utilised to improve retrieval performance [Bandyopadhyay et al., 2012, Massoudi et al., 2011]. This technique leads to significant improvements over the TREC Microblog track 2011-2014 in several works, as mentioned in [Ounis et al., 2011, Soboroff et al., 2013, 2012]. However, these methods are heavily based on external resources such as Google or Bing searches, where, if the original query is weak, the returned results can introduce more noise into the expansion



process. Alternatively, the gap caused by the length of microblog documents can be bridged by expanding a tweet containing a URL using crossbedding information (such as the title of the web page) [Efron et al., 2012, El-Ganainy et al., 2014]. In recent years, this kind of query expansion technique has commonly been deployed in TREC Microblog tracking. Exploiting the information from the content linked by the URL can improve retrieval performance, but it will also increase the computational costs to fetch the URL information. Another approach to improving microblog retrieval performance is the integration of query expansion with external evidence (such as Probase, Freebase, and Wikipedia) to understand the query [Martins et al., 2016, Wang et al., 2017].

Many microblog retrieval studies have utilised PRF for query expansion techniques [Berendsen et al., 2013, Chen et al., 2018, Chy et al., 2019, Lau et al., 2011, Martins and Callan, 2018, Metzler et al., 2012, Miyanishi et al., 2014, Zingla et al., 2016]. An earlier study, proposed in [Miyanishi et al., 2013], manually selects tweets, then estimates the relevance feedback for the selected tweets. In [Miyanishi et al., 2013], the authors proposed a graph-based model that could generate a storyline for a given query within the PRF framework. To highlight the short-term importance of a given query, Albakour et al. [2013] used PRF to extend the user information needs to address the sparsity challenge. Another approach used a global knowledge base (such as Freebase or Probase) to bridge the gap of weak semantic similarity in microblogs [Wang et al., 2017]. Alternatively, Fan et al. [2015] detected the underlying entities in the original query and then applied these in its relevance feedback model. In practice, utilising global evidence (such as a knowledge base) alongside the PRF framework requires a double run for a query and can increase the computational efficiency [Carpineto and Romano, 2012]. Most of the above contributions hold the same assumption as classic PRF: that the initial retrieved documents are relevant to the original query.

Temporal information has been widely implemented in previous microblog retrieval research. Dong et al. [2010] proved that time is important to capture relevance information. To explore the relationship between time and relevance, Li and Croft [2003] purposed a temporal language approach by combining time information and relevance models. Efron and Golovchinsky [2011] integrated temporal indications from the initial retrieved documents to decide the rate parameter for the query's likelihood model. They then applied PRF to calculate the expansion features. Liang and de Rijke [2015] utilised data fusion techniques to propose a burst-aware model that fused retrieved document lists from different retrieval systems for a given query with temporal evidence to boost microblog search performance. Due to the natural differences of the microblog data compared with the long documents, Hasanain and Elsayed [2017] tested and proposed a variety of query predictions with the ability to improve the microblog search by predicting a given query. To identify users' behaviours using recent features, Choi and Croft [2012] blended temporal information from the PRF with the relevance model to reformulate the original query. They selected a period based on social user features (e.g., retweets) to derive relevant tweets that would be utilised to extend the original query. Relevant evidence in a real application (including microblog retrieval) points to cluster collectively in time (i.e., event). Based on this concept, Efron et al. [2014] offered a retrieval version for microblog searches that implemented temporal comments to measure the relevant information density. To dynamically filter real-time tweets, Tan et al. [2016] took advantage of the top-ranked tweets from previous days and employed a dynamic emission method in TREC 2015. In this thesis, we aim to enhance the efficiency of the pseudo-relevance feedback performance based on selecting the informative training subset.

## 2.3 Unsupervised Learning

Machine learning strategies can be categorised into four main themes: supervised, unsupervised, semi-supervised, and reinforcement learning. Supervised methods are learning from a set of examples as the input value (called the label training set instance) and the corresponding output, in this kind of learning, can be produced as binary classes. These classes can belong to the right label known as a classification problem. The label training set in supervised learning is often provided by a human. Some common supervised learning approaches include classification, regression, support vector machine, and random forest. In previous research, it has widely been used with text-based applications. However, in unsupervised learning algorithms, the strategy draws an inference from the collections, including input data instances without labeled data instance. It is clear that unsupervised learning techniques aim to find the structure of the input data without utilising explicit labels provided by a human. Some common algorithms include the topic model and clustering. This chapter will review one of the most critical unsupervised learning algorithms: topic modeling.

### 2.3.1 Topic Modeling

In unsupervised learning techniques, a topic model is a kind of statistical model for finding the patterns of correlated information (topics) that appear in a set of documents. These documents include web pages, news articles, and web posts, such as tweets. Probabilistic latent semantic analysis (PLSA) [Hofmann, 1999] and LDA [Blei et al., 2003] are probabilistic topic models, and have been employed widely in understanding text corpora. In particular, because LDA extends PLSA through increasing Dirichlet priors on the distributions of the topic, LDA is a far superior model of text generation. Over the last decade, quite a few versions have been developed for LDA and PLSA on

account of their extensibility. Variants include dynamic topic model [Blei and Lafferty, 2006], author-topic-community model [Li et al., 2015a], author-topic model [Rosen-Zvi et al., 2004], and the social topic model [Cha and Cho, 2012]. These models are structured to deal with normal texts but also include extra features like authorship and social links and associations.

Canonically, the LDA model (a hierarchical parametric Bayesian model) offers a means of topic mining within a large set of documents. Specifically, documents in an LDA model are a mixture of topics where a topic is identified in terms of words as a probability distribution in the vocabulary set in the test collection. This probability distribution is then learned through statistical inference, so that words are linked to a topic and their distribution over documents. Generally, LDA-like models work by groping relevant words semantically, in a holistic topic, by using words' co-occurrence evidence at the document level [Wang and McCallum, 2006]. As a result, they are highly sensitive to the length and number of documents associated with each topic. This emerging issue has become common in recent year. The next section discusses the limitations of the topic model with short text documents.

### **2.3.1.1 Topic modeling for social media content**

In modern society, rich information is passed on through short texts. This applies to a broad range of web portals, instant messaging, online marketing, email communication, and social media. Typically, these texts are informal, short in length, and noisy. Understanding these unannotated texts offers an efficient way to obtain important highlights from large text collections. Standard human capacities cannot handle the massive breadth of this data, hence the need for efficient and robust techniques. When it comes to machine-based discovery of thematic information by text mining from a massive set of documents, topic modeling has proved particularly adept. This approach

takes documents as a mixture of probabilistic topics distributed over words [Blei et al., 2003]. Some dormant structures in a document collection are uncovered by these topics and can be understood by inferential statistical models. Typical topic modeling approaches, like LDA, have demonstrated considerable success with large documents; however, these approaches fall short when it comes to analysing smaller short documents [Hong et al., 2011, Zhao et al., 2011]. Crucially, shorts texts (e.g., a tweet) offer limited word co-occurrence information when compared to longer documents [Wang and McCallum, 2006].

Nonetheless, as contextual information is highly sparse, short text documents still challenge the ability of traditional approaches (which were mainly designed to mine longer texts) to reveal information. In addition, the highly imbalanced document distribution characteristic of short texts continues to create challenges. Traditionally, a principal objective for topic modeling approaches has been to maximize the probability of the data; however, such models appear to lose efficiency for rare or extraordinary topics [Jagarlamudi et al., 2012]. As a result, in extrinsic tasks (e.g., document classification, term similarity tasks), the performance of these topic models may not be adequate [Chang et al., 2009].

As a result of the low word count in short texts, these models can fail to produce an accurate picture of the interrelation of given words. When the topic distribution over a set of documents is skewed, LDA-based approaches appear to acquire more common topics contained in most of the documents instead of scantily available topics in fewer documents. Recent research suggests that if topic distribution over documents is profoundly skewed, it will be challenging to identify topics from fewer documents through LDA [Tang et al., 2014]. Indeed, to detect hot trends occurring in real-time over social media, or for recently emerging events discovery, focusing on rare topics is imperative [Chen et al., 2013a].

To handle the sparsity issue, two important heuristic strategies have been endorsed within the context of short texts. The first assumes that a short text document is about one latent topic. The strategy was employed in early approaches to topic models, like the mixture of unigrams [Nigam et al., 2000]. Though this assumption may not work effectively with large documents, it is an appropriate fit for some short texts and may be useful in addressing elements of the sparsity issue [Zhao et al., 2011]. The second strategy capitalizes on several heuristic connections between short text extracts and cumulates them into large pseudo-documents before applying a standard topic model. This strategy is widely employed on social media platforms. For instance, such contextual information is amassed through hashtags, time, authorship, and locations linked to social media posts and these cues are helpful for aggregation [Hong and Davison, 2010, Mehrotra et al., 2013, Weng et al., 2010]. However, more general types of short texts cannot be handled easily by this strategy. For example, this strategy struggles with search queries that are lacking useful ties but are extensively observable in various aspects.

In modeling short texts, the ineptitude of LDA has been addressed in many recent studies. For instance, closely associated short texts can be amassed into pseudo documents prior to training the topic model [Weng et al., 2010]. Alternatively, those models focusing on external knowledge (e.g., Wikipedia, Freebase, or Probase) can be employed to assist in inferencing topics contained in short texts [Phan et al., 2008]. Alongside these instances, a number of arbitrary versions of LDA have been broached to address the need to analyse particular short texts [Chen et al., 2013a, Chong and Chua, 2013, Zhao et al., 2011]. Dissimilar to the data-based or task-based approaches discussed above, the emergence of topic models that focus on generally relevant short text is also under consideration. A unique combination of unigrams, known as a biterm topic model, is proposed by [Yan et al., 2013] to augment short text topic modeling.

It is effective in handling short extracts. Nonetheless, it is not based on LDA and is a unique type of topic model with a mixture of unigrams. Thus, the downsides observed with LDA-based techniques are not addressed by a biterm model, but its highly limited flexibility is also noted. Finally, the dual-sparse topic model changes LDA specifically for short text topics and specified terms for each such topic [Lin et al., 2014].

In available literature, much research has focused on sparse short texts, and a large portion of previous research has primarily focused on augmenting the density of data using relevant information. For instance, Hong and Davison [2010] developed topic models on accumulated tweets that have the same word. The study reported higher efficiency for these models compared to those produced directly for original tweets. For short text, a search-snippet-based measure of similarity is proposed by [Sahami and Heilman, 2006]. Similarly, Jin et al. [2011] employ modal long-text data to discern topics in short text documents by transferring learning from these auxiliaries. A different approach is to use relevant topic models to handle short text data sparsity. For document topic distributions and distribution of topic-term carried out in short text topic modeling, Lin et al. [2011] proposed sparse constraints to deal with sparse short texts.

Contrarily, concerning topic imbalance, LDA performance augmentation is acquired by making use of already available information to lead the progress of topic learning [Andrzejewski et al., 2009, Jagarlamudi et al., 2012] or by employing asymmetric Dirichlet prior to document-topic distribution [Wallach et al., 2009]. It is noted that, in practical terms, often it is not known what knowledge a given collection contains in its underlying structure; therefore, the acquisition of prior information is a difficult task. For different applications, discovering adequate parameter estimations for asymmetric Dirichlet priors is a daunting task that depends on a scenario. Here, it is also assumed that LDA (and its different versions) can still stay flexible with the

help of symmetric Dirichlet priors. Coherent topics are discovered by using general lexical knowledge [Chen et al., 2013b]. Thus, LDA topic imbalance can be practically improved with symmetric Dirichlet priors, and it is an approach that is highly needed. In short, the aforementioned modalities are not independent of their scenarios nor are they easily extendable in IR tasks as the proposed models in the thesis.

## 2.4 Summary

This chapter provided the background for IR and machine learning from the fundamental principles, starting with the retrieval models. Then, we showed the development of IRT followed by machine learning (especially unsupervised learning approaches) and discussed the correlation with IR. The next section covered information analysis in social media, including an overview of the benefits and challenges of social media in terms of extracting relevant information. Then, we discussed variation in microblog retrieval models and the techniques used to find relevant information to satisfy users' information needs. Different applications were discussed, including summarisation, topic and event detection, personalisation, and opinion retrieval. The rest of this section shows the development of query expansion techniques in terms of improving the search effectiveness. Finally, topic modeling approaches were introduced, including the development of applied topic modeling with the microblog context.

This chapter has reviewed the recent literature in the area of microblog search. The task of discovering relevant information in short texts (such as tweets) is still struggling to distinguish representative features for a given user's information need. In order to fill this gap, this thesis utilised well-established technologies in information retrieval and integrated them with unsupervised learning areas for application to microblog search. The next chapters introduce the proposed models for microblog search in detail and



the experiments' evaluation of the proposed models.



## Chapter 3

# Topic Aware Pseudo-Relevance Feedback for Boosting Microblog Search

---

### 3.1 Introduction

Recently, many users use microblog applications (such as Twitter) to find information relevant to their needs. Improving microblog search has received much attention in recent years through the application of classic retrieval models or the utilisation of external resources to augment the original user's information needs (more details discussed in the Section 3.3.3). However, such evidence is not always available to use, and the most important concern is the increasingly high volume of published microblog documents, such as tweets. Early in Section 2.2.1, we show the main microblog characteristics, including time sensitivity, short length, informal writing style, and redundant information. These limitations can increase vocabulary or term mismatching within the microblog context.

In this chapter, the proposed model is based on a well-known methodology for managing vocabulary mismatch: automatic query expansion. This procedure can

be automated through pseudo-relevance feedback. Pseudo-relevance feedback (also called implicit feedback or blind feedback) is a method intended to assume what the user may find relevant without having explicit user feedback. Pseudo-relevance feedback assumes that the top results have higher accuracy and features those results that are expected to show the user's research topic.

The proposed model hypothesizes that the initial retrieved documents can include relevant information at its latent topics that can be useful for improving the retrieval effectiveness to meet user information needs (Related to **RQ1**). This chapter presents a pseudo-relevance feedback model, including relevance feedback and topic-based query expansion for microblog retrieval. The proposed model combines the lexical and topical evidence from pseudo feedback with respect to the original query. The significant benefit of the proposed model is stability and robustness as it does not need external resources to expand the original query. A general framework of the proposed model is shown in Figure 3.1. This chapter has been published in [Albishre et al., 2017].

The remainder of this chapter is structured as follows: Section 3.2 presents preliminaries overview of the related works, Section 3.3 discusses the proposed model and Section 3.5 concludes the proposed model.

## **3.2 Preliminaries**

### **3.2.1 Latent Dirichlet Allocation (LDA)**

LDA is a classic generative probabilistic model that assumes given documents are distributions over topics and each topic is distributions over words. It generates a mixture of topics utilising word co-occurrences at the documents level. While each

topic represents a set of words, the probability distribution of a word  $w_i$  in a document  $d$  is estimated as follows:

$$P(w_i|d) = \sum_{j=1}^V P(w_i|z_j) \times P(z_j|d) \quad (3.1)$$

where  $P(w_i|z_j)$  denotes the multinomial distribution probability over all words  $w_i \in z_j$ ,  $P(z_j|d)$  denotes the topic weight for a given document  $d$ ,  $P(z_j)$  denotes the topic assignment of topic  $z_j$  and  $V$  is the number of topics.

The LDA output includes a set of latent topics,  $Z$ . Each topic  $z_j$  is represented by a multinomial distribution over a set of words, described as  $\phi_j = \{\varphi_{j,1}, \varphi_{j,2}, \dots, \varphi_{j,n}\}$  where  $\varphi_{j,h}$  is the probability of a word  $w_i$  in the topic  $z_j$ , and the sum of all elements in the topic space is described as  $\sum_{h=1}^n \varphi_{j,h} = 1$ . For all topics,  $Z$ , over all words in a document,  $\Phi = \{\phi_1, \phi_2, \dots, \phi_V\}$  is the composition for each topic  $z_j$ . Each document is represented by a multinomial distribution over  $Z$  topics as  $\Theta_d = \{\vartheta_{d,1}, \vartheta_{d,2}, \dots, \vartheta_{d,V}\}$  where  $\vartheta_{i,j}$  indicates the proportion of topic  $z_j$  for a given document  $d$ , and the sum of all elements in  $\Theta_d$  is denoted as  $\sum_{j=1}^V \vartheta_{d,j} = 1$ . In this thesis, Gibbs sampling is used to estimate the posterior distribution for LDA inference [Porteous et al., 2008].

### 3.2.2 Language Model

**Query Likelihood:** This thesis used a language model on its proposed model's frameworks. The lexical evidence is an important part of understanding the text in microblog contents. The lexical evidence with the language model is a special case of probabilistic retrieval, and the state-of-the-art model for the language model is query likelihood. The query likelihood model proposed by Ponte and Croft [1998] assumes that the

probability of the relevance model  $P(Q|d)$  can be generated using the probabilities of query features  $Q$  given document  $d$ . Thus, documents are ranked based on the posterior probability  $P(d|Q)$  using Bayes rule:

$$P(d|Q) \propto P(Q|d)P(d) \quad (3.2)$$

where  $P(d)$  is the prior probability that  $d$  is relevant to any query and  $P(Q|d)$  is the query likelihood of the given document  $d$ .

The multinomial query likelihood model  $P(Q|d)$  is described as follows:

$$P(Q|d) = \prod_{i=1}^{|Q|} P(w_i|d) \quad (3.3)$$

where  $|Q|$  is the number of query's terms and  $P(w_i|d)$  is the relevance model that computes the probability of word  $w_i$  based on its distribution in document  $d$ .

Different smoothing techniques are proposed in the existing IR literature, and an effective technique is presented in Zhai and Lafferty [2004]. They used Bayesian smoothing for their language model by using Dirichlet priors, as follows:

$$P(w|d) = \frac{|d|}{|d| + \mu} \cdot \frac{c(w, d)}{|d|} + \frac{\mu}{|d| + \mu} \cdot P(w|\mathcal{C}) \quad (3.4)$$

where  $c(w, D)$  is the word frequency in the document,  $P(w|\mathcal{C})$  is the probability of the collection language model, and  $\mu$  is the smoothing parameter for  $\mu \in [0, +\infty)$ .

The final form of query likelihood using Dirichlet prior smoothing is described in Zhai and Massung [2016], as follows:

$$P(Q, d) = \sum_{w \in Q, d} c(w, Q) \log \left( 1 + \frac{c(w, d)}{\mu \cdot P(w|\mathcal{C})} \right) + |Q| \log \frac{\mu}{\mu + |d|} \quad (3.5)$$

where  $c(w, d)$  is the word count in a given document  $d$ ,  $c(w, Q)$  is the word count in a given query  $Q$ ,  $|d|$  and  $|Q|$  are the respective lengths of the document and query, and  $P(w|\mathcal{C})$  is the probability of the word in the collection that is used to normalise the model.

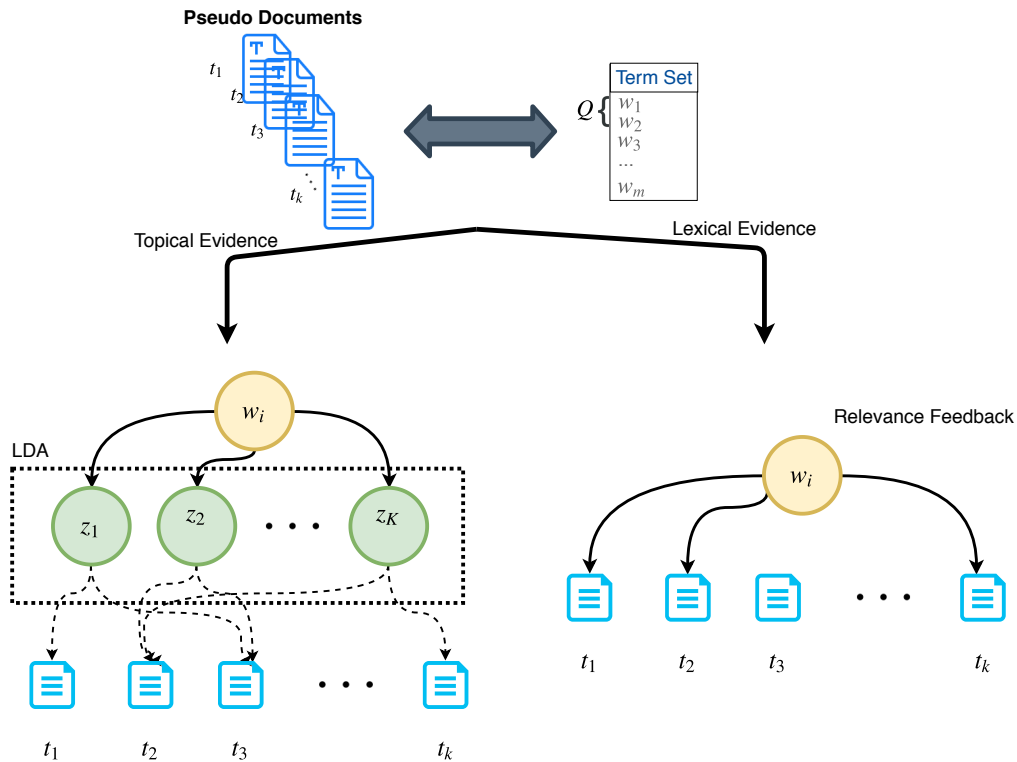
**Pseudo-Relevance Feedback:** There are different pseudo-relevance feedback techniques proposed in the literature. All of them seek to enhance the retrieval process while avoiding the vocabulary mismatching problem. One of the more robust models is RM3 [Abdul-Jaleel et al. [2004], Lavrenko and Croft [2001]], in which the basic idea is to estimate the relevance feedback using relevance models such as query likelihood, BM25. Then, after the relevance feedback has been estimated, it interpolates with the original query. Lv and Zhai [2009b] proved that RM3 was the most effective and robust of a number of state-of-the-art query expansion models.

$$P(w|R) = \sum_{D \in \mathcal{D}} P(D) P(w|D) \prod_{i=1}^n P(q_i|D) \quad (3.6)$$

where  $\mathcal{D}$  is a set of feedback documents,  $P(D)$  often assumes to be uniform that can be ignored and  $\prod_{i=1}^n P(q_i|D)$  is query language model.

$$P'(w|R) = \gamma P(w|R) + (1 - \gamma) P(w|Q) \quad (3.7)$$

where query model  $P(w|Q)$  is the original query model.



**Figure 3.1:** The proposed model framework.

### 3.3 Topic aware PRF framework

Vocabulary mismatch is a central challenge faces microblog search (as discussed in the introduction). This issue arises when the users' interest is not enough to represent the relevant documents. Different techniques can be used to solve this problem, including query expansion. Therefore, selecting discriminative expansion features will improve retrieving relevant documents to meet user's interest needs.

As shown in the previous sections, pseudo relevance feedback (PRF) using automatic query expansion relies on the assumption that expansion terms found in the top retrieved documents (i.e., unlabelled feedback set) can be used to boost the original



user query. In order to overcome the limitations of PRF in the microblog context, the proposed model (TAPRF) identifies expansion terms from different evidence levels within the initial retrieved tweets set. In the first stage, it takes the advantages of topic modeling (e.g., LDA) to extract more discriminative terms from the first-pass ranked documents. Then, lexical evidence from the relevance feedback for the pseudo documents is extracted. In this way, there are two extracted features sets, from the previous steps, that are integrated into one set through liner combination. Finally, the top features from the integrated features set are used to expand the original query. The proposed model is illustrate in Figure 3.3.

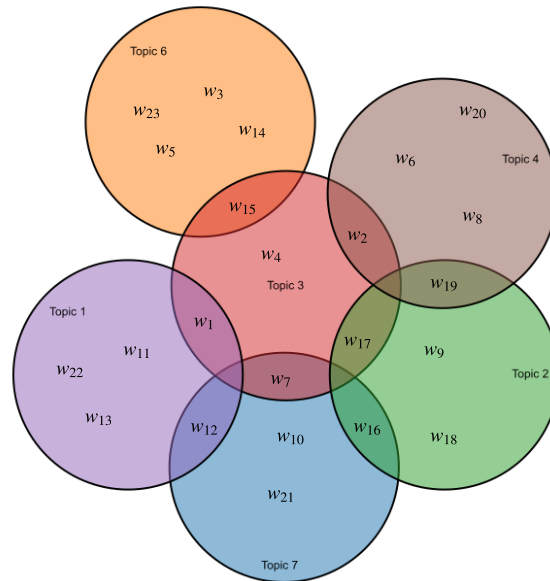
Initially, let  $Q$  be a user query  $Q = \{q_1, q_2, \dots, q_n\}$  that issue at specific time following the TREC microblog dataset formate and  $\mathcal{C}$  be a tweet collection where  $t$  is a tweet. A ranking model  $rel(Q, t)$  is applied to individual tweets to obtain the initial retrieved tweets, denoted as  $F$ , as follows:

$$rel(t, Q) = \begin{cases} \sum_{q \in Q} rank(t, q) & , \text{ if } q \in t \\ 0 & , \text{ if } q \notin t \end{cases} \quad (3.8)$$

where  $rank(t, q)$  ranks a tweet the collection based on their relevance to the given user query. This method can also be adapted to any retrieval model, including BM25, RFD, TF-IDF or a language model. This thesis utilises the language model through query likelihood with Dirichlet smoothing. As shown in Section 3.2.2,  $rank(t, q)$  used Equation 3.2 where document notion is equivalent to tweet  $t$ .

### 3.3.1 Infer topical evidence

The initial retrieved tweets are the most relevant tweets for the given user query from a collection. In the case of PRF, the critical task is how to select the most relevant



**Figure 3.2:** Topic Example for the top 50 documents in MB2011

information from the top-ranked documents; however, there are implicit relations between terms in a set of pseudo feedback that could carry semantics representations. In this stage, we utilise LDA to discover the topic representation at the ranked documents level. We discovered latent topics from a set of initial retrieved tweets (i.e., pseudo tweets). Therefore, the main objective of this stage is to maximize the discovery of relevant informative features, using the topical features from a set of pseudo feedbacks.

Importantly, we discovered latent topics in pseudo feedback set  $F$  using LDA. The result from LDA, as mentioned in Section 3.2.1 consists of a set of latent topics  $Z$ , each topic  $z_j$  is represented by a multinomial distribution over words and each tweet  $t$  in the pseudo document set a multinomial distribution over topics represents  $F$ . Figure 3.2 illustrates a real example of six topics produced using LDA for the top ranked tweets set for a given query. It is clear from the figure that there is some association between topics in some words that could to indicate the relevant association between

these words.  $Score_{LDA}(w)$  estimates the topic distribution in the pseudo-documents set  $F$ . In this stage,  $Score_{LDA}(w)$  is defined as the average of the topic distributions of tweets that include a given the word  $w$  in  $F$ .

### 3.3.2 Estimate the relevance feedback

Relevance feedback models have been shown to be very effective in improving the search system performance with different techniques as well as applications (as explained in Section 3.3). Typically, feedback can involve a user to get explicit actions. The user can make a judgment as whether each document returned in the results is useful or not. These decisions from the user produce the relevance judgment; however, in the case of microblogs, the process of user interaction to get the relevance judgment is not practical for several reasons. First, the high velocity of published tweets can overwhelm the user with incoming information. Also, the freshness of the published tweets could cause user judgment to drift from their needs. This research holds the same assumption as PRF methodology, which assumes the initial ranked results are relevant.

As shown in the previous section, latent evidence is discovered from the top retrieved tweets for a given query. In this section, the same evidence input utilised in the previous section is used to find the lexical evidence through relevance feedback. Relevance models for lexical evidence try to weigh the term based on its dependency on the document (i.e., tweet), where it considers the tweet weight from the ranking model. Where the LDA does not consider the tweet weight for a given query that gained from the ranking model, it is assumed that a word is generated from finite topics (as shown in the previous section). The common relevance feedback models in IR and IF assume the word is generated directly from a set of documents that could be implicit or explicit feedback. The proposed model assumes a word can generate from

different levels of evidence including the topical and lexical.

$$Score_{rel}(w) \propto \sum_{t \in F} p(w|t) * p(t) * rel(t, Q) \quad (3.9)$$

where  $F$  is the top pseudo feedbacks set,  $p(w|t)$  is the probability distributions of word  $w$  as in Equation 3.1,  $rel(t, Q)$  is estimated as in Equation 3.8 and the tweet prior  $p(w|t)$  is often assumed to be uniform.

### 3.3.3 Integration the evidence

Then, after the relevance feedback  $Score_{rel}(w)$  is estimated, as in Equation 3.9, we integrate the lexical evidence via relevance model  $Score_{rel}(w)$  with the topical evidence via topic model  $Score_{LDA}$ . The combination  $Score_{TAPRF}(w)$  is done using liner interpolation, as follows:

$$Score_{TAPRF}(w) = (1 - \lambda) * Score_{rel} + \lambda * Score_{LDA}(w) \quad (3.10)$$

where  $\lambda \in [0, 1]$  is the something parameter. This step is done to improve the performance of the relevance model estimation.

The final step of the proposed model is a linear computation for the new expansion query words between the original query model and the evidence levels model  $Score_{TAPRF}(w)$  and computes, as follows:

$$p(w|\theta'_Q) = (1 - \gamma) * \frac{f(w, Q)}{|Q|} + \gamma * Score_{TAPRF}(w) \quad (3.11)$$

where  $\gamma \in [0, 1]$  is a parameter to balance the using of pseudo-relevance feedback.

**Algorithm 1:** Score\_TAPRF()

---

**Input:** a set of initial ranked pseudo documents  $F$ ;  
number of topics  $V$ ;  
a control parameter  $\lambda$ ;

**Output:** a scoring function  $Score_{TAPRF}(w)$ ;

- 1  $W = \{w | w \in t, t \in F\}$ ;
- 2 // estimate topical evidence
- 3 Generate  $V$  topics  $Z$  by applying LDA to the top ranked pseudo documents  $F$   
;
- 4 **foreach**  $w \in W$  **do**
- 5 | assign LDA weight to  $w$  using  $Score_{LDA}$ ;
- 6 **end**
- 7 // estimate lexical evidence
- 8 **foreach**  $w \in W$  **do**
- 9 | **foreach**  $t \in F$  **do**
- 10 | | **if**  $w \in t$  **then**
- 11 | | |  $Score_{rel}(w)$  as in Equation 3.9;
- 12 | | **end**
- 13 | **end**
- 14 **end**
- 15 // integration the incoming evidence
- 16 **foreach**  $w \in W$  **do**
- 17 | **return**  $Score_{TAPRF}(w) = (1 - \lambda) Score_{rel}(w) + \lambda Score_{LDA}(w)$ ;
- 18 **end**

---

### 3.4 Algorithms

Algorithm 1 describes the process of the proposed model, where the input includes a set of the initial ranked pseudo-documents  $F$ , the required number of LDA topics  $V$  and a control parameter  $\lambda$ . The output is a scoring function that estimates the evidence from the topical and lexical level. The algorithm starts with the initialization for the vocabulary set from a given pseudo-documents set  $F$  at Step 1. Then, as in Step 3, it generates topics  $Z$  for a set of pseudo-documents  $F$  by utilizing topic model LDA. Then, it assigns the LDA weight to each word  $w$  in the vocabulary set  $W$  from Step

4 to 6. To estimate the lexical evidence, as the proposed model used the relevance model, it calculates the weight for a given word  $w$  as in Equation 3.9 from Step 8 to 14. Finally, it integrates the words' weight that comes from the LDA and relevance model in Step 17.

The topical and lexical evidence estimation mainly determines the time complexity of the Algorithm 1. For topical evidence's time complexity, it decided by LDA that it is linear with the number of documents and topics [Wei and Croft, 2006]. Where each iteration for LDA is  $\mathcal{O}(1)$ , a fixed small set of incoming tweets and number of topics were utilised in the proposed model. The time complexity of LDA is determined by the number of documents and topics, and can be linear  $\mathcal{O}(F \times V)$ , where  $F$  is the number of pseudo-documents and  $V$  the number of topics. For each given query, the proposed model often used the number of pseudo-documents between 50 to 10 tweets where the number of topics is about seven. In the lexical evidence's time complexity, for each word  $w$ , it takes  $\mathcal{O}(F \times L)$  where  $L$  is the average length of the tweet. Therefore, the time complexity of the Algorithm is  $\mathcal{O}(F \times L \times V)$ .

### 3.5 Summary

This chapter presents the details of the topic-aware pseudo-relevance feedback model for microblogs (TAPRF). TAPRF exploits the topic model in the initial retrieved tweets to generate topical evidence. Another evidence space is discovered from the same set of tweets with a relevance model that represents the lexical evidence. Then, TAPRF integrates the extracted features set to re-weight each feature and re-formulate the original query. Applying the TAPRF approach to TREC microblog collection 2011-2014 shows that the proposed model is significantly improved compared with the lexical based methods. The evaluation results of the proposed model will present in

Chapter 6. Though, the proposed model is sensitive to the given retrieved tweets due to the variation of user query quality. This issue is modeled through automatic pseudo test set adaption to reduce the uncertain information caused by PRF's assumption that first pass documents are relevant for all queries. In the following chapter, we propose a model that aims to detect the most likely pseudo test collection automatically.





## Chapter 4

# Discovery of Informative Training Set for Effective Microblog Search

---

### 4.1 Introduction

The Topic Aware Pseudo-Relevance Feedback introduced in Chapter 3 can discover latent evidence from the income ranking documents and expose topical information to improve PRF within microblog data. PRF via query expansion is a method for utilising initial retrieved documents to improve document retrieval performance. It assumes that the first pass retrieved documents contain relevant information. The significant advantage is that PRF does not require any human judgment. The idea behind PRF is that top-ranked documents obtained in an initial query-based retrieval are likely relevant to the user's information needs. Then, the selection of some terms or features from those top-ranked documents are used to expand the original query and likely enhances the retrieval effectiveness.

On the other hand, the initial ranked documents consist of both relevant and irrelevant documents. Expanding the initial user need from top-ranked documents in a real

application such as microblog search could introduce more noise features in order to address language difficulties (for example, mismatched vocabularies). Microblog also have unique characteristics that distinguish them from longer documents, including time sensitivity, short lengths, unstructured phrases and insufficient information (as shown in Chapter 2). The hypothesis is that the first-pass retrieved documents include relevant information to a given query, but the proportion of relevant information is different from one query to another. Thus, using expansion features from the initial ranked documents without assessing these documents can be harmful to the retrieval performance as it can increase the noise features.

However, this assumption does not always hold in the microblogosphere [Chen et al., 2010, Miyanishi et al., 2013] due to the overwhelming quantity of noise and redundant information it contains. For example, many irrelevant microblog documents overlap with relevant information, as they share the same terms or features. The top-ranked documents obtained by a given query include many irrelevant documents that contain some query terms. Therefore, the use of all the top-ranked documents as a set of relevant documents cannot significantly improve retrieval performance [Lv and Zhai, 2010].

As we mentioned, in top-ranked documents, a document can be either relevant or non-relevant, and a PRF's performance is strongly related to how much the higher-ranked documents are relevant to a user's information needs. If this assumption in the PRF is incorrect, it may lead to query drift [Carmel and Yom-Tov, 2010]. Thus, a significant obstacle to determining PRF performance is how to select a high-quality set of documents in the first-pass retrieval before applying the PRF process. The relationship between top-ranked documents has remained unexploited because they are in a query-oriented order [Lee et al., 2008]. Consequently, the quality of the selected expansion terms strongly depends on the variety of the top-ranked documents

[Zhai and Massung, 2016]. This can be done by conducting a series of experiments to determine the right number of top-ranked documents. However, this methodology is hard for real application (such as microblog retrieval), as the performance of an information retrieval (IR) model heavily depends on the input data collection.

To overcome the limitations of classic PRF when applied to microblogs, we propose a model to dynamically estimate a number of pseudo-documents that are used to select expansion features for a given query (related to **RQ2**). The proposed model, TBS, automatically selects the best  $k$  pseudo-documents from the first-pass retrieved documents as a random variable rather than fixed- $k$  (related to **C2**). The proposed model views the topical distribution of the features in each candidate document's subset using a topic model technique. We assume that the proposed model improves the query expansion for microblog search performance regarding dynamic pseudo-document selection.

The proposed model contains two main phases. For a given query, it automatically determines the number of pseudo feedbacks used in the relevance feedback, based on the initial retrieved documents set that is divided into small subsets. In each subset, we infer the discriminative power of features that included in the subset to see whether the subset is suitable to use as pseudo feedbacks. In the proposed model, we utilise latent Dirichlet allocation (LDA) to discover the latent topics in the subset. LDA is used as it has the ability to capture the latent relationships between features, whereas this is difficult using term-based approaches (such as TF-IDF and BM25). When the proportion of relevant documents is high, some focused topics appear from the initial retrieved documents set. Using the discovered latent topics in each candidate subset, we determine the best  $k$  value by calculating the precision. In the next phase, from the selected pseudo-documents in the previous step, we integrate the discovered topical distribution features with lexical evidence in the relevance model concerning

its temporal distribution (related to **C3**). A general framework of the proposed model is shown in Figure 4.1. This chapter has been published in [Albishre et al., 2018].

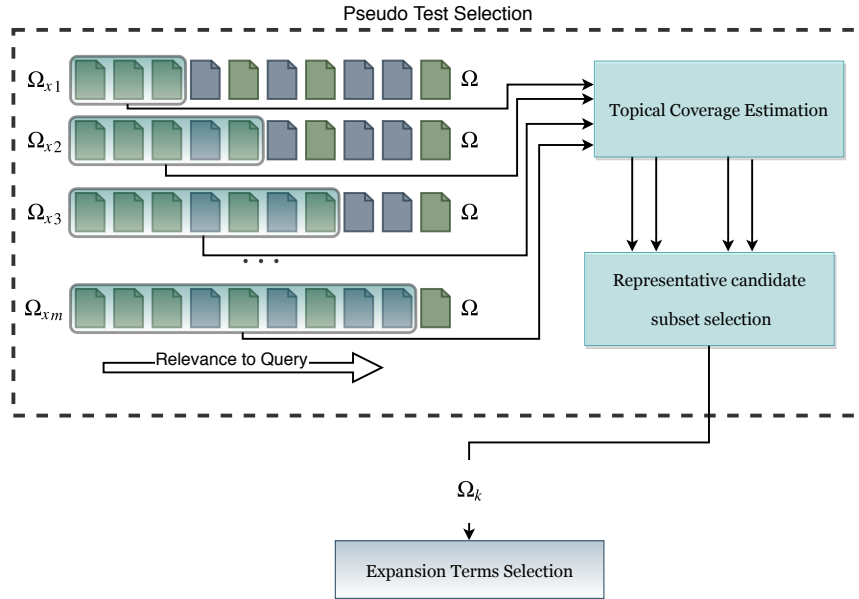
## 4.2 The Proposed Model (TBS) Framework

Given a query  $Q$  at time  $T$  and a microblog collection  $C$ , a retrieval system will return a ranked document set that represents the first-pass retrieved list. The input of the proposed model is a top-ranked documents set with chronological information for a given query. The input is estimated as in Equation 3.8. Then, it determines the more likely relevant feedback, including the representation of the latent topics and automatically documents feedback selection (described in section 4.2.1). Finally, from the selected feedback, we exploit the relevance model’s feedback features with topical features weight (described in section 4.2.2).

### 4.2.1 Pseudo Feedback Selection

A typical pseudo-relevance feedback assumes that top-ranked documents in the first-pass retrieval are relevant regardless of different queries. Then, it expands the original query using the top-selected documents as feedback. In modern retrieval applications (such as microblogs), the PRF performance for a specific query is often sensitive to the adequate number of feedback documents. In this stage, we introduce a fully automatic query-specific feedback document selection model in the top-ranked documents.

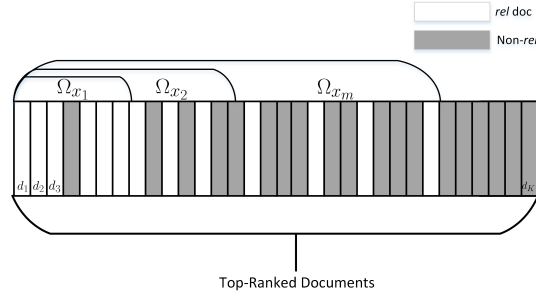
The main concern is how to capture the relevant information (e.g., terms, topics, or themes) from first-pass retrieved documents. The relevant information has more focus on the searched topic and is isolated from irrelevant information [Lv and Zhai, 2009a]. Many common features may be shared between relevant and irrelevant information. A



**Figure 4.1:** TBS general architecture

classic retrieved model, such as term-based has a limited ability to infer the relevant term in microblogs. Few focus topics appear in relevant documents where topics are diverse in irrelevant documents (as shown in the previous chapter). This observation reinforced the possibility of determining relevant topics from the first-pass retrieved documents set because the frequency of focused topics seems to be higher than in diverse topics.

For a given query, we assume that the top-ranked documents in the initial retrieval are a good indicator to retrieve relevant documents. Ideally, the precision of top-ranked documents is strongly related to the value of  $k$ . For example, if the precision of the top 30 of query “A” is 0.863 and the precision of the top 30 of query “B” is 0.15, using a fixed number of top-ranked documents as feedback for all queries can reduce the retrieval performance. The question now is how to decide parameter  $k$  to make  $\Omega_k$



**Figure 4.2:** Selecting pseudo-relevance feedback

have the highest precision.

#### 4.2.1.1 Pseudo Documents Construction

To decide a suitable value for parameter  $k$  for a given query  $Q$ , let  $\Omega = \{t_1, t_2, \dots, t_{|\Omega|}\}$  be the top-ranked documents that contains a set of words  $W = \{w_1, w_2, \dots, w_n\}$  and  $n$  is the total number of unique words. Then, we assume  $\Omega$ 's subsets,  $\Omega_{x_1}, \Omega_{x_2}, \dots, \Omega_{x_m}$  where  $1 \leq x_1 \leq x_2 \leq \dots \leq x_m \leq |\Omega|$ . The candidate subset  $\Omega_{x_j}$  is the top- $x_j$  ranked documents in  $\Omega$ , such that  $\Omega_{x_j} \subseteq \Omega$ . Figure 4.2 shows the process of selecting the top- $k$  pseudo-relevant documents. Table 4.1 shows an example of how to build each candidate subset  $\Omega_{x_j}$  from a give set of ranked tweets  $\Omega$ .

**Table 4.1:** An Example of subset constrain

Candidate subset	Tweets	Words
$\Omega_{x_1}$	$t_1, t_2, t_3, t_4$	$w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8$
$\Omega_{x_2}$	$t_1, t_2, t_3, t_4, t_5, t_6$	$w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8,$ $w_9, w_{10}, w_{11}, w_{12}$
$\Omega_{x_3}$	$t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8$	$w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8,$ $w_9, w_{10}, w_{11}, w_{12}, w_{13}, w_{14}$

Since manual judgment is always time consuming, it is almost impossible to judge

all the documents with regards to a query. Therefore, we assess documents in the candidate subset  $\Omega_{x_j}$  using the topic model LDA to select the more reliable candidate subset that contains more likely relevant documents in the next step.

#### 4.2.1.2 Candidate Subset Topical Coverage Estimation

Due to the absence of relevance judgment, a document in the initial retrieved documents can be relevant or irrelevant to the user's information need. The probability distribution  $P(w_i|\Omega_{x_j})$  indicates the degree for a word  $w_i$  where relevant words usually substantially related to the topics spaced together. Based on this intuition, we believe that the top- $x_j$  documents in  $\Omega_{x_j}$  are possibly relevant. As mentioned in Section 3.2.1, based on the LDA, we estimate the probability distribution of a word  $P(w_i|t)$  in tweets by using Equation 3.1. Thus, estimating the word probability from a higher level (such as a set of retrieved documents rather than an individual document) can retrieve the relevant word, which appears from latent topics. The word probability  $P(w_i|\Omega_{x_j})$  is estimated as follows:

$$P(w_i|\Omega_{x_j}) = \frac{\sum_{w_i \in t, t \in \Omega_{x_j}} P(w_i|t)}{|\{t|t \in \Omega_{x_j}, w_i \in t\}|} \quad (4.1)$$

After estimating each word  $w_i$  in the candidate subset  $\Omega_{x_j}$ , we rank all words included in the subset. Then, we can select the top words in the candidate set to represent the given subset. In the proposed model, we took all words in the given subset that describe the topical evidence.

### 4.2.1.3 Representative Candidate Subset Selection

We view the probability distribution  $P(w_i|\Omega_{x_j})$  in each candidate subset  $\Omega_{x_j}$ . The reflection of the relevant information in a candidate subset  $\Omega_{x_j}$  is represented by a discriminative power of word distribution. To achieve this, we obtain the topical probability distribution  $P(w_i|\Omega_{x_j})$  for each word  $w_i$  in the candidate subset  $\Omega_{x_j}$  as in Equation 4.1. While the number of features in the  $\Omega_{x_j}$  is observed and the proportion of relevant feedback is unexplored, the increase of extraneous features in the next subset  $\Omega_{x_{j+1}}$  indicates that more uncertain information can generate to the relevance feedback model.

**Definition 1 (representative candidate subset)** : *Let  $\Omega$  be a set of ranked tweets for a given query. A candidate subset  $\Omega_{x_j}$  that is  $\Omega_{x_j} \subseteq \Omega$  and  $|\Omega_{x_j}|$  is candidate subset size is representative if the total of the words that include in the subset has covering weight is higher than other subsets. Representative candidate subset is defined as:*

$$\operatorname{argmax}_{1 \leq x_j \leq m} \sum_{i=1}^n \frac{P(w_i|\Omega_{x_j})}{|\Omega_{x_j}|} \quad (4.2)$$

where the probability distribution  $P(w_i|\Omega_{x_j}) = 0$  if  $w_i \notin \Omega_{x_j}$ .

Through this function, each candidate subset is represented using the topical coverage over a set of topics. Therefore, they can analyse both the coverage of the topic and the relevance from a topic space. In addition, the representative candidate subset can be fed into a relevance features discovery or simply searched for relations between expansion words. In the next stage, we use the selected subset words regarding reformatting the user information needs.



### 4.2.2 Expansion Terms Selection

After selecting the candidate subset  $\Omega_k$  as in Section 4.2.1, this phase aims to estimate the weight for word  $w_i \in \Omega_k$ . We integrate the topical weight  $P(w_i|\Omega_k)$  to the relevance model  $P(w_i|R)$ . We follow the literature [Abdul-Jaleel et al., 2004, Lavrenko and Croft, 2001] for estimating the relevance model. The proposed relevance model computes the weight for a word  $w_i \in \Omega_k$ , as follows:

$$P(w|Q) \propto P(w|\Omega_k) + \sum_{d \in \Omega_k} P(w|d)P(d) \prod_{i=1}^n P(q_i|d) \quad (4.3)$$

where  $P(w|\Omega_k)$  denotes the topical word distribution of word  $w$  in the candidate subset  $\Omega_k$  (estimated as in Equation 4.1),  $\prod_{i=1}^n P(q_i|d)$  indicates the query likelihood language model with Dirichlet smoothing for a given document  $d$  and the document prior  $P(d)$  is often assigned a uniform status. A microblog timestamps are not uniform, users often favour recent microblogs for a given query. Based on this, the recency-based document  $P(d|T_d)$  is utilised in this chapter to integrate the temporal information following Li and Croft [2003], as follows:

$$P(d|T_d) = r * e^{-r*(T_Q - T_d)} \quad (4.4)$$

where  $r$  is the parameter that controls the temporal information,  $T_Q$  is the query issue time and  $T_d$  is the document publication time.

The final step of the proposed model is a linear combination of the relevance model  $P(w|Q)$  and the original query model  $\theta_Q$ . We computed it as follows:

$$P(w|\theta_{Q'}) = \lambda P(w|\theta_Q) + (1 - \lambda) P(w|Q) \quad (4.5)$$

where  $\lambda \in [0, 1]$  is a parameter to balance the using of pseudo-relevance feedback. Then, we compute the simple form for the original query model as follows:

$$P(w|\theta_Q) = \frac{c(w, Q)}{\sum_{w' \in V} c(w', Q)} = \frac{c(w, Q)}{|Q|} \quad (4.6)$$

where  $c(w, Q)$  is the count of term  $w$  in  $Q$ , and  $|Q|$  is the length of the query.

In summary, we estimate the relevance model for each word  $w$  in the selected feedback and integrate with the topical distribution weight. This combination can give the discriminative terms an appropriate weight. Additionally, we consider the temporal distribution of each document in the selected documents set. To meet the initial user information needs, we interpolated the oriental query model with the relevance model to select high-quality expansion terms.

### 4.3 Algorithms

Algorithm 2 describes the process of pseudo-documents selection, where  $\Omega$  is the initial top-ranked documents for a given query  $Q$  and a subsequence  $x_1, x_2, \dots, x_m$  of  $1..n$ . The algorithm starts with the initialization from Step 1 to Step 3 including the subsets numbers. In each candidate subset  $\Omega_{x_j}$ , it uses the LDA to generate three representation levels including a set of topics  $Z$ , the documents-topics proportion  $\Theta$  and the topics-words probability distribution  $\phi$  in Step 5. From Step 8, it also sums each word distribution  $P(w|\Omega_{x_j})$  as calculated in Equation 4.1, and then assigns all

---

**Algorithm 2:** selectK()

---

**Input:** top-ranked documents  $\Omega$ , a subsequence  $x_1, x_2, \dots, x_m$  of  $1..n$ **Output:** top- $k$  documents  $\Omega_k$ 

```

1 let  $j = 1, j_0 = j$ ;
2 let  $E$  is an empty vector;
3  $W = \{w|w \in d, d \in \Omega\}$ ;
4 while  $j \leq m$  do
5     Generate topics  $Z$  by applying LDA to  $\Omega_{x_j}$ ;
6     let  $s = 0$ ;
7     foreach  $w_i \in W$  do
8         |  $s += P(w_i|\Omega_{x_j})$ ;                               // based on Eq. 4.1
9     end
10     $E[j] \leftarrow s/(|\Omega_{x_j}|)$ ;
11    if  $E[j] > E[j_0]$  then
12        |  $j_0 = j$ ;
13    end
14     $j = j + 1$ ;
15 end
16 return  $\Omega_{x_{j_0}}$ ;

```

---

word distributions in  $E$ . In the last step, it selects the highest value of  $E$  and returns  $\Omega_k$  in Step 16.

## 4.4 Summary

In this chapter, we proposed a model to improve microblog search by determining an informative training set. The proposed model automatically estimates the representative pseudo feedbacks that are utilised in the relevance feedback model using the topical distribution for the initial ranked documents for a given query. In addition, we integrated the topical distribution information from the selected documents into the relevance feedback model to infer the discriminative power for each feature. To

demonstrate the proposed model's effectiveness, empirical experiments were done on standard TREC microblog 2011-2014 datasets compared with state-of-the-art baseline models in Chapter 6. The experimental outcomes concludes that the proposed model outperformed all baseline models with significant improvement for microblog search.

## Chapter 5

# Query-based Unsupervised Learning for Improving Social Media Search

---

### 5.1 Introduction

The quantity of relevant information in the initial retrieved set relies on the original user information needs; if the query is short or vague, uncertain information can impede the following selection process. Applying PRF to social media texts without considering the nature of these texts (e.g., time sensitivity or short length) can introduce more noise features [Chen et al., 2018, Miyanishi et al., 2013, Wang et al., 2017]. The need to improve social media search has received much attention in recent years as discussed in the previous chapters. Thus, it is challenging to reduce noise resulting from frequent terms for a query-based unsupervised method.

In term of statistics, unsupervised learning intends to infer prior probability distributions  $p(x)$  and supervised learning intends to infer conditional probability distributions  $p(x|Y)$  for any input object  $x$  based on a large training set  $Y$ . Priors can be

created using a number of statistical methods (e.g., a normal distribution) or determined from previous experiments; however, in real applications, priors are universal if the relevant background is not taken into account. In this research, we consider the relevant background by using a query ( $Q$ ); therefore, **unsupervised learning** in this research intends to infer a probability distribution  $p(x, Q)$ .

We depart from existing methods by observing that the lack of word co-occurrence information in short texts has the main impact on improving the social media search. The ultimate aim is to capture optimal implicit relationships from the initial retrieved tweets in order to infer more knowledge to serve user needs. As mentioned previously, the central problem is how to reduce uncertainties in retrieved tweets, as we do not know which tweets are relevant to the user's needs.

This chapter proposes a new query-based unsupervised method to overcome the limitations of the aforementioned issues when deriving high-quality terms for retrieved documents (related to **RQ3**). We, first, receive the initial results for a given query and then select the top-ranked tweets. Using the top-ranked tweets, we build a new documents space, based on a query-based tweets-pooling strategy, to discover a new relationship between the user information needs and the selected tweets (related to **C3**). From this new space, we model the associations between a proposed entity, including the original query, top-ranked tweets, and the intermediate set (related to **C3**).

Then, we obtain a novel weight for each word in the selected tweets in respect to their implicit relevance information (related to **C3**). Therefore, we believe that the proposed model will be useful for conducting high-quality unsupervised learning in order to find high-quality text features. Details of the models are described in the following sections. This chapter has been published in [Albishre et al., 2019].

## 5.2 Problem Formulation

Given a query  $Q$  and a tweets collection  $C$ , an information retrieval system returns an initial ranked list of tweets  $T$  that contain  $k$  tweets. The proposed model utilises the top- $k$  ranked tweets, which may include both relevant and non-relevant tweets for training the model. A tweet in the ranked list may be relevant to query  $Q$ ; however, it may be non-relevant to what users want. Let  $\Omega$  be a set of all words in  $T$  where  $w \in \Omega$  is a tweet token (e.g., a word). For each tweet  $t_x \in T$ , we assume there is a probability function  $P_r : T \rightarrow [0, 1]$ , which shows the probability of the tweet's relevance to what users want. For a given information retrieval system, which predicates the probability of relevance of tweets and sorts them in a ranked list, we have the following property:

$$P_r(t_1) \geq P_r(t_2) \geq \dots \geq P_r(t_k).$$

The research problem is how to select and weight words  $w \in \Omega$  for describing the relevant knowledge about what users want based on the given query  $Q$  and the retrieved tweets  $T$ . This is a challenging task because the relationship between  $\Omega$  and  $Q$  is a many-to-many relation. In turn, a reasonable latent relation is very hard to derive because  $Q$  is very small and the intermediate set  $T$  between  $\Omega$  and  $Q$  contains uncertain tweets that may be relevant or non-relevant. The proposed model will propose a method to reduce the uncertain information in the retrieved tweets. The relationship between  $\Omega$  and  $Q$  can then be derived reasonably. The selected words will also be used as a new alternative representation of the initial user query  $Q$  to improve the social media search with high-quality relevant information.

### 5.3 The Proposed Model (QUSTM)

In this section, we show the proposed model to be used with social media short texts. The core contribution of this chapter, as discussed in the introduction and problem formulation has three main components: we exploit a new tweets-pooling schema and model its relationships. Based on the complex representation, we interpret each discovered feature to describe its discriminative power.

#### 5.3.1 Latent Relationships

The main input for this phase is a set of retrieved tweets  $T$  for a given query  $Q$ , which is used by a user to describe what she/he wants. Each tweet  $t_x \in T$  is considered as an unlabeled tweet. In the proposed model, we use a language model, “the query likelihood model with Dirichlet smoothing” [Zhai and Lafferty, 2001b] to get the retrieved tweets  $T$ . This can be adapted for use with any retrieval model, such as BM25 [Robertson et al., 1994], pattern-based PTM [Zhong et al., 2010], or RFD [Li et al., 2015b]. Let  $\Omega$  be a set of words (text features) for describing the relevant knowledge contained in retrieved tweets  $T$ . The objective here is to select  $\Omega$  from  $T$  based on the query  $Q$  in order to describe the relevance to the user’s need.

To solve this challenging task, we are going to discuss the relationship between  $\Omega$  and  $Q$  through an intermediate set, retrieved tweets  $T$ . The obvious relationship between  $Q$  and  $T$  is a set-valued mapping that is defined, as follows:

$$\Gamma : Q \rightarrow 2^T \quad (5.1)$$

where mapping  $\Gamma$  can generate  $m$  sub-sets of tweets. We call each sub-set a *virtual document*.



**Table 5.1:** An example of virtual documents.

Tweet	Content
$t_1$	$w_1, w_2, q_1, w_3, w_4, q_2, w_5$
$t_2$	$w_1, w_2, w_6, w_4, w_8, q_2$
$t_3$	$w_2, w_4, w_9, q_1, w_{10}, w_{12}$
$t_4$	$w_{11}, w_7, q_2, w_9, w_8, w_{10}$
$t_5$	$q_1, w_1, w_2, w_6, w_4, w_{11}$
$t_6$	$w_8, q_2, w_4, w_8, w_{10}, w_{12}$

Virtual Document	Content
$\Gamma(q_1)$	$t_1, t_3, t_5$
$\Gamma(q_2)$	$t_1, t_2, t_4, t_6$

**Definition 2 (Virtual Document)** : Let  $Q = \{q_1, q_2, \dots, q_m\}$  be a given query. A virtual document is a set of tweets that are related to an aspect of query  $Q$ . Formally, for each virtual document, there is a query term  $q_j$ , such that, the virtual document can be denoted as  $\Gamma(q_j) = \{t_x | t_x \in T, q_j \in t_x\}$ .

Table 5.1 shows an example of how to build virtual documents where the original query is  $Q = \{q_1, q_2\}$  and a set of initial retrieved tweets is  $T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$ . In this example, we have two virtual documents (i.e.,  $\Gamma(q_1)$  and  $\Gamma(q_2)$ ). A virtual document  $\Gamma(q_1)$  includes all tweets in the initial retrieved documents  $T$  that include query term  $q_1$  and is defined as  $\Gamma(q_1) = \{t_1, t_3, t_5\} = \{q_1, q_2, w_1, w_2, w_3, w_4, w_5, w_6, w_9, w_{10}, w_{11}, w_{12}\}$ .

The rationale for making use of the virtual documents rather than original tweets when extracting informative features from the retrieved tweets is two-fold. First, in the above discussion, we use a mapping  $\Gamma$  to generate  $m$  virtual documents  $\Gamma(q_j)$  for all  $q_j \in Q$ . For a given document, people (human beings) usually decide the relevance of the given document when reading through the whole document; in most cases, we

say a document is relevant if we find a relevant sentence or paragraph in the document. Thus, if any tweet  $t_x \in \Gamma(q_j)$  is relevant, then we believe that  $\Gamma(q_j)$  is relevant. Based on the above discussion, we can define:

$$P_r(\Gamma(q_j)) = \max\{P_r(t_x) | t_x \in \Gamma(q_j)\}$$

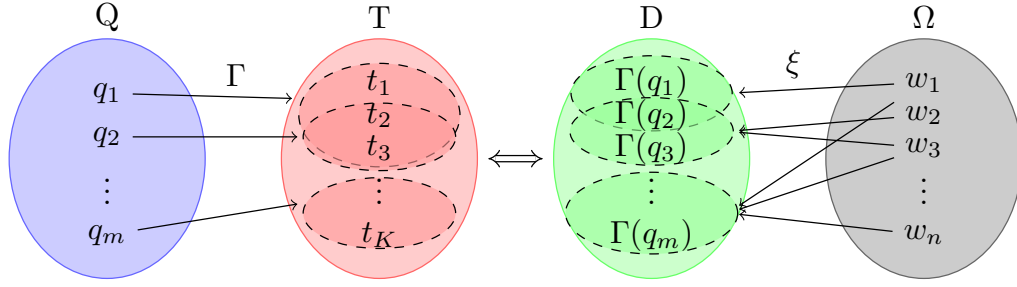
Then, we can easily prove that:

$$\text{mean}(P_r(\Gamma(q_j))) \geq \text{mean}(P_r(t_x))$$

This conclusion states that mapping  $\Gamma$  can reduce the extent of uncertainty in the retrieved tweets.

Since the association between query terms is weak (especially with short text) the generation of a new space, such the proposed virtual documents, can increase the number of associations between the query terms and related terms. For example, as shown in Table 5.1, tweets  $t_1, t_3$  and  $t_5$  overlap only in  $\{q_1, w_2, w_4\}$ . Thus, the only associations that can be generated from the tweets terms are based on the proposed virtual document definition. As shown in Table 5.1, a virtual document  $\Gamma(q_j) = \{t_1, t_3, t_5\}$  includes more tweet terms where the number of associations is increased (e.g.,  $w_{10}$  and  $w_{11}$ ) in the same virtual document. In this manner, our proposed virtual document schema has reduced the gap of the association between query terms and candidate terms where  $\Gamma(q_j) \cap \Gamma(q_{j+1}) \neq \emptyset$ .

After obtaining the virtual documents for a set of retrieved tweets  $T$ , a new document space is introduced in which the relationships between  $\Omega$  and the new document space should be investigated. Let  $D = \{d_1, d_2, \dots, d_m\}$  be the set of virtual documents



**Figure 5.1:** The relationship between  $Q$ ,  $\Omega$  and intermediate sets.

$D = \{\Gamma(q_j) | q_j \in Q\}$ . We can now obtain a one-one relation between  $Q$  and  $D$ , that is:

$$q_j \in Q \Leftrightarrow d_j = \Gamma(q_j) \in D$$

We can also obtain a mapping between  $\Omega$  and  $D$ , as follows:

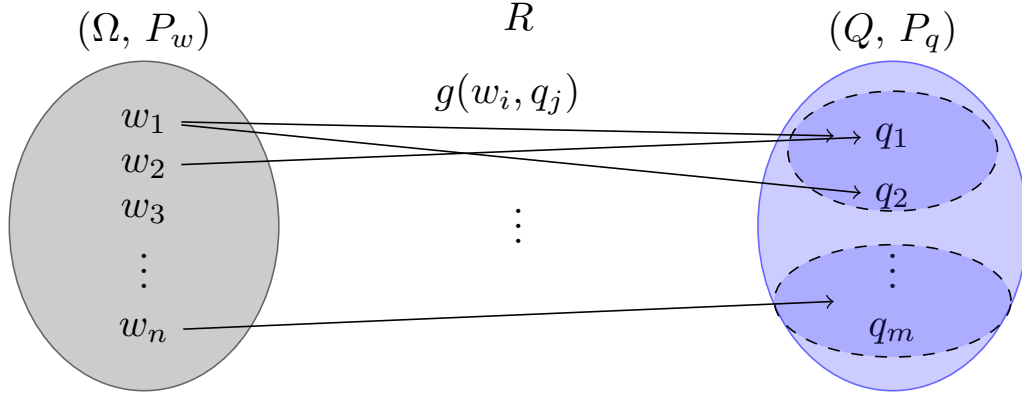
$$\xi : \Omega \rightarrow 2^D \quad (5.2)$$

where  $\xi(w) = \{d_j | d_j \in D, w \in \Gamma(q_j)\}$ . Figure 5.1 shows the relation between  $Q$  and  $\Omega$  through the intermediate sets.

Based on the above analysis, we can describe the relationship between  $\Omega$  and  $Q$ , as follows:

$$R : \Omega \rightarrow 2^Q \quad (5.3)$$

where  $R(w) = \{q_j | q_j \in Q, w \in \Gamma(q_j)\}$ . Figure 5.2 shows the relationships between  $\Omega$  and  $Q$  in detail. The relationship includes  $P_q$ , a probability function for describing query terms' specificity,  $P_w$ , a probability function for describing the relevance of words to query  $Q$  and  $g(w_i, q_j)$  which describes the strength of word  $w_i$  related to



**Figure 5.2:** The relationship between  $\Omega$  and  $Q$  via  $R$ .

query term  $q_j$ .

### 5.3.2 Term Estimation

The main obstacle of the proposed model to determine the relevance of a word is the absence of relevant guidance, such as real user-relevant feedbacks. In the previous section, we assume that there are weak implicit relationships that can be strengthened through aggregating tweets into virtual documents. In this section, we show the mechanism that estimates the probability of observing a word  $w_i$  through a score function  $Score(w_i)$  in the virtual documents space  $D$  to a given user need  $Q$ . A score function  $Score(w_i)$  can be used to calculate a representative weight for each word  $w_i$  for all  $w \in \Omega$ , as follows:

$$Score(w_i) = P(w_i, D, Q) \cdot P_w(w_i) \quad (5.4)$$

where the joint probability  $P(w_i, D, Q)$  estimates the probability of relevance of the observing the word  $w_i$  in the virtual documents  $D$  and  $P_w(w_i)$  is an uncertainty factor

that is used to deal with uncertainty in virtual documents.

To compute the joint probability  $P(w_i, D, Q)$ , we estimate the expected value of a word  $w_i$  over the virtual documents  $D$ , as follows:

$$\begin{aligned}
 P(w_i, D, Q) &= P(Q) \cdot P(w_i, D|Q) \\
 &= P(Q) \cdot \sum_{q_j \in Q} [P(q_j) \cdot P(w_i, D|q_j)] \\
 &\propto \sum_{j=1}^m P(w_i, D|q_j) \cdot P(q_j)
 \end{aligned} \tag{5.5}$$

where the score that estimated by the joint probability  $P(w_i, D, Q)$  is a proportional probability of a word  $w_i$ 's relevance and the probability  $P(Q)$  assumes uniformity overall words.

The following final estimation, for the score function  $Score(w_i)$  of a words  $w_i$ , is given when we substitute Equation 5.5 into Equation 5.4:

$$Score(w_i) = P_w(w_i) \cdot \sum_{j=1}^m P(w, D|q_j) \cdot P(q_j) \tag{5.6}$$

In the implementation, we also give the following definitions for the concepts in Section 5.3.1. To instantiate the joint probability  $P(w_i, D, Q)$  from Equation 5.5, we estimated two main components: the word strength in a given virtual document  $P(w_i, D|q_j)$  and the query terms' specificity  $P(q_j)$ . First, we estimate the strength of word  $w_i$  to query term  $q_j$ , as follows:

$$P(w_i, D|q_j) = g(w_i|q_j) = \frac{tf(w_i, q_j)}{|\Gamma(q_j)|} \tag{5.7}$$

where  $tf(w_i, q_j)$  is a term frequency of  $w_i$  in  $\Gamma(q_j)$  and  $|\Gamma(q_j)|$  is the size of a virtual document  $\Gamma(q_j)$ , that indicates the number of tweets in  $T$  with query term  $q_j$ .

Second, we estimate the query terms' specificity  $P(q_j)$  for a given  $q_j$ , as follows:

$$P(q_j) = \frac{k - |\Gamma(q_j)|}{k} \quad (5.8)$$

where  $k$  is the number of tweets in the initial retrieved tweet set  $T$ .

Since the virtual documents are constructed from unlabeled tweets, it is necessary to take into account the uncertainty inherent in the weighting function. The probability  $P_w(w_i)$  is used to deal with the underlying uncertainty in the relevance estimation in Equation 5.6. We estimate the number of query terms  $P_w(w_i)$  that map it in their virtual document between  $Q$  and  $\Omega$ , as follows:

$$P_w(w_i) = |R(w_i)| = |\{q_j | q_j \in Q, w_i \in \Gamma(q_j)\}| \quad (5.9)$$

Please note that  $P_w(w_i)$  and  $P(q_j)$  can be normalised as a total probability function. Thus, for information retrieved or ranking, we can ignore the totals as they are constant for all terms or query terms.

Finally, after estimating the weight for each word  $w_i$  in  $\Omega$ , we ranked all words  $w \in \Omega$  based on their weight. Then, we selected the top words to represent user information need, denoted as  $Q'$ .

### 5.3.3 Algorithms

Algorithm 3 shows the proposed model framework where the input contains a set of retrieved tweets  $T$ , the original query  $Q$  and a word  $w_i$  in  $\Omega$ . The algorithm starts with

**Algorithm 3:** Score( $w_i$ )

---

**Input:** A set of ranked tweets  $T$ , a query  $Q$ , a word  $w_i$   
**Output:** weight for  $w_i$

```

1 // construct virtual documents
2 foreach  $q_j \in Q$  do
3    $\Gamma(q_j) = \emptyset$ ;
4   foreach  $t_x \in T$  do
5     if  $q_j \in t_x$  then
6        $\Gamma(q_j) = \Gamma(q_j) \cup t_x$ ;
7     end
8   end
9 end
10 // calculate  $P(w_i, D, Q)$  based on Eq.(5.6)
11 foreach  $q_j \in Q$  do
12   estimate  $P(w_i, D|q_j)$  as in Eq. (5.7);
13   estimate  $P(q_j)$  as in Eq. (5.8);
14    $w'_i = P(w_i, D|q_j) \cdot P(q_j)$ ;
15    $P(w_i, D, Q) = P(w_i, D, Q) + w'_i$ ;
16 end
17 calculate  $P_w(w_i)$  as in Eq. (5.9);
18 return  $P_w(w_i) \cdot P(w_i, D, Q)$ ;

```

---

the initial steps for contracting the virtual documents, from step 2 to step 8, where it aggregates all tweets in  $T$  that contain a given query term. Then, it uses the virtual documents to discover the implicit relationship between  $(\Omega, P_w)$  and  $(Q, P)$  for a given word  $w_i$ . The algorithm used its a novel weighting schema; it starts from step 10-14 of the algorithm by verifying the given word  $w_i$ , then estimates the word  $w_i$  frequency in the current virtual document, as in Equation 5.7, multiplied by the virtual document frequency, as in Equation 5.8. The algorithm then continues for each virtual document  $\Gamma(q_i)$  that contains a word  $w_i$ . Finally, it generalises a given word  $w_i$  based on its frequency it is in the new space overall (in our case, in the virtual documents). Building the virtual documents can be done once, after which the weight for each word  $w_i$  in  $\Omega$  can be estimated.

The time complexity of Algorithm 3 is determined by the “foreach” loops. The time complexity of the first “foreach” loop is  $\mathcal{O}(m \times k \times L)$  where  $L$  is the average size of a tweet. The time complexity of the second “foreach” loop depends on the process used when estimating  $g(w_i, q_j)$ , and the time complexity is  $\mathcal{O}(m \times S)$  where  $S$  is the average length of a virtual document  $\Gamma(q_j)$  and  $S = \mathcal{O}(k \times L)$ . So, the time complexity is  $\mathcal{O}(m \times k \times L)$ .

## 5.4 Summary

Unsupervised learning for social media data has been widely utilised in a number of short-text applications, including information retrieval, text summarisation, topic discovery, and events detection. In this chapter, we propose a query-based unsupervised learning method that aims to capture the implicit relationships that can increase the search performance of social media short-text by coping with the sparsity problem. The core aim of the proposed model is to reduce uncertain information in the given tweets. Extensive experiments show the effectiveness of the proposed model coupled with a state-of-the-art language model, probabilistic model, and temporal and topic baseline models over the TREC microblog datasets, as shown in Chapter 6. The proposed model is a breakthrough for unsupervised learning in this research area.



## Chapter 6

### Experiments

---

In this chapter, a set of experiments is conducted to verify three main hypotheses in this research. These hypotheses are:

1. To reduce noisy information from the extracted features, the proposed proximity model can incorporate the semantic features with the lexical evidence.
2. Not all users' queries are well-formulated. Thus, considering all top-retrieved documents as a set of relevant feedbacks (as in PRF-based models) for use in expanding the original user query can reduce the retrieval performance.
3. To reduce the insufficiency of statistical evidence in short text documents (such as tweets) self-augmentation of tweets in the first-pass results, based on query terms distribution into virtual space, can reduce the uncertainty of finding relevant information for a given user's need.

To evaluate these hypotheses, this chapter describes the proposed methods' experimental evaluation. This chapter discusses the testing environment, including datasets, baselines models, and methods' performance results. For the first hypothesis, the

proposed model results and discussions correspond to the major key finding: it shows significant improvement, in comparison to baseline methods, based on effectiveness criteria.

For the second hypothesis, the proposed model achieved significant results in terms of efficiency and effectiveness compared to the baseline methods. More findings and discussion on the selection criteria of a dynamic set of pseudo feedback for a given user query compared with fixed selection mechanism-based methods were presented.

For the third hypothesis, it reports the results and the discussions for the proposed approach that, in terms of effectiveness, outperforms the state-of-the-art baseline models. In addition, we discuss the results from all proposed models in addition to the top automatic runs reported in TREC Microblog tracks for all datasets used in our evaluation.

The hypotheses proposed in this study have been evaluated based on microblog retrieval application. Well-established TREC microblog datasets have been used as test collections for the proposed models. To assess how the search output satisfied the user information need, typical evaluation metrics for IR have been utilised in addition to a statistically significant test.

## 6.1 Data Collections

The experiments are carried out using the TREC 2011-2012 datasets (known as Tweets2011 [Ounis et al., 2011]) and the TREC 2013-2014 datasets (known as Tweets2013 [Soboroff et al., 2013]). Table 6.1 presents the collections statistics in detail. The size of the dataset Tweets2011 is approximately sixteen million tweets from a period of two weeks from January 23, 2011, to February 8, 2011 and covering significant occasions

**Table 6.1:** The statistics of test collections

Collection	Size	Days	Topic Set	Queries
Tweets2011	16M	16	MB2011	49
			MB2012	60
Tweets2013	240M	60	MB2013	60
			MB2014	55

such as the Us Super Bowl and the Egyptian Revolution. The Tweets2013 collection is much larger, at approximately two hundred forty million tweets, and includes a span of two months from February 1, 2013, to March 31, 2013.

Different kinds of tweets exist in both datasets (including retweets and replies), which results in considerable noise. In addition, the tweets were published in various languages. Each day, the corpus is split into files called blocks, each of which contains approximately 10,000 tweets compressed using gzip. Each tweet is in JSON format, as shown in Figure 6.1.

The Tweets2011 dataset has topic sets that consist of 49 (MB2011) and 59 (MB2012) topics. Tweets2013 has topic sets that consist of 60 (MB2013) and 55 (MB2014) topics. Each official topic includes the topic number, title and topic timestamp, as shown in Figure 6.2. In this thesis, we utilised all the topic sets for Tweets2011 and Tweets2013. The NIST assessors applied a standard pooling method for assessment by assigning multi-scale judgments to every tweet indicated as highly relevant, relevant, and not relevant.

The relevant judgment is a multi-scale that ranks tweets as highly relevant, relevant, and not relevant. Figure 6.3 shows the relevance ratio over all test sets. NIST provides the researchers an API<sup>1</sup> to crawl the Tweets2011 microblog dataset. For Tweets2013,

<sup>1</sup><https://github.com/lintool/twitter-gear/>

```
{
  "text": "I hate when I try to be slick but then I get
  caught",
  "id_str": "28966361187225601",
  "id": 28966361187225601,
  "created_at": "Sun Jan 23 24:04:53 +0000 2011",
  "retweeted": false,
  "retweet_count": 0,
  "favorited": false,
  "user": {
    "id_str": "139196085",
    "id": 139196085,
    "screen_name": "DanaAngeline",
    "name": "Danaaaaa"},
  "requested_id": 28966361187225601}
```

**Figure 6.1:** An example of tweet structure

```
<top>
<num> Number: MB009 </num>
<title> Toyota Recall </title>
<querytime> Tue Feb 08 21:41:26 +0000 2011 </querytime>
<querytweettime> 35090855064764416 </querytweettime>
</top>
```

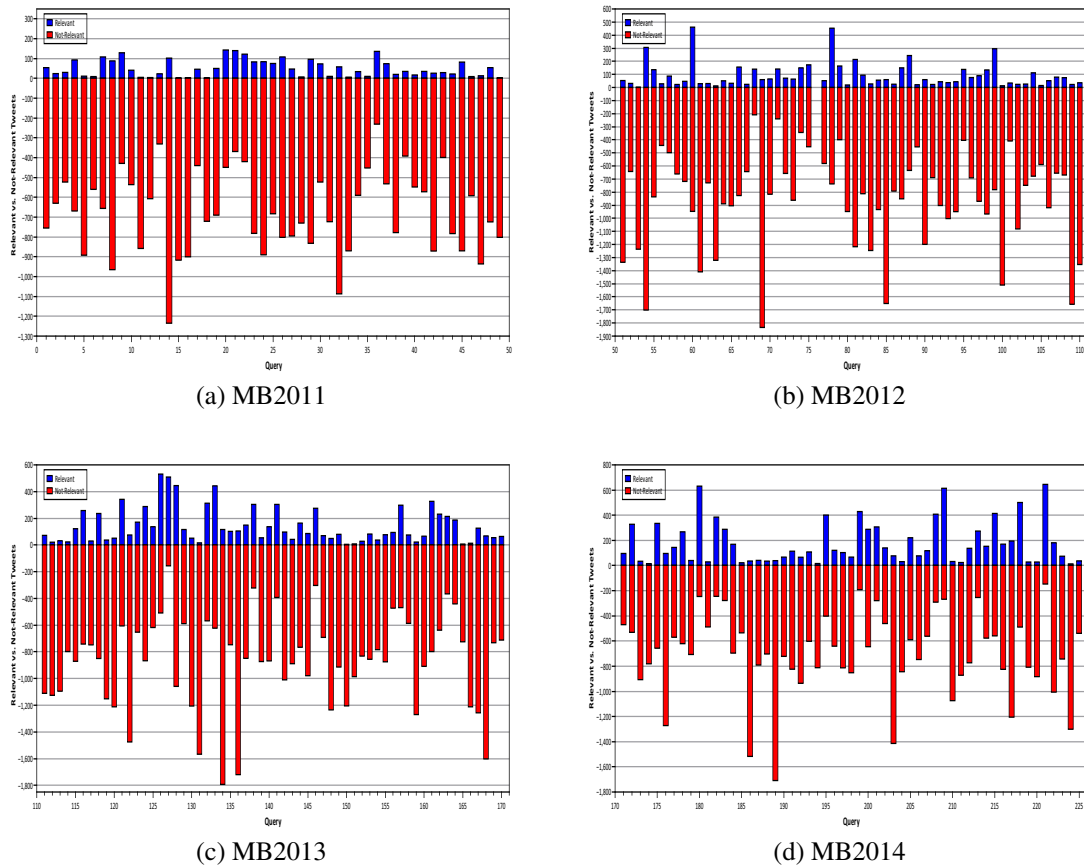
**Figure 6.2:** An example of query in the test set

we are utilising the official API, and there is a downloadable version available in this repository<sup>2</sup>. We are utilising those APIs and received local copies for the datasets.

### 6.1.1 Preprocessing

Text cleaning requires attention to numerous issues. Microblog text content frequently contains misspelt words, incorrect punctuation, erratic spacing, and other irregular

<sup>2</sup><https://github.com/castorini/Tweets2013-IA/>



**Figure 6.3:** relevanc Judgment Ratio

features. Punctuation generally compares to the use of phoneme features in spoken language; to depend on delimited sentences framed by surprising punctuation can be exceptionally dangerous. In some corpora, conventional prescriptive standards are regularly overlooked.

The main purpose of document preprocessing is to decrease the dimensionality to control the number of terms in the document [Li et al., 2015b]. Moreover, the cleaning stage will improve performance and efficiency by making the data uniform. Different cleaning methods widely used in text mining are stop word removal, ignoring short terms, special character removal, and stemming. Preprocessing documents is a critical

phase to enhance the retrieval model and improve retrieval performance.

Stop words typically point to the most widely-recognised words in a language. One of the challenges in natural language processing (NLP) is that no universal stop word list exists. Every language has its list, and each may change over time. For instance, English has more than one list because the communication behaviour between people has changed over time.

Words such as “is”, “which”, “the”, “and”, and “of” add noise to text data and, thereby, reduce the efficiency of text-data documents. Therefore, stop-word removal plays an important role in document pre-processing steps. Significantly, stop word removal can both reduce the noise for the text document and maintain the core term in documents to make processing more efficient and effective.

Stemming is another useful technique. The main purpose of stemming is to cut words down to their root. For example, in English, the words “smoker” and “smoking” have the same root stem: “smoke”. The literature contains different algorithms for this method. The Porter stemming [Porter, 1980, Willett, 2006] algorithm is one popular technique. We treated the tweets and text queries based on the following steps:

- Tweet filtering: non-English tweets make data noisy, so we discarded these tweets, using a language detector called *ldig*<sup>3</sup>, in cases where the Tweets2011 dataset did not have a “lang” attribute.
- Retweet treatment: we normalised the tweets that start with “RT”.
- We filtered out tokens (in text and hashtags) that include non-ASCII characters, including emojis and symbols. For hashtags, we tokenised the hash symbol and kept the tokens since they may contain query related keywords.

---

<sup>3</sup><http://github.com/shuyo/ldig>

- We removed the stop words in tweet text using stop word list, and then stem word using the Porter algorithm.
- Any tweets with less than two tokens were ignored.

All the above steps followed the TREC microblog guidelines [Ounis et al., 2011, Soboroff et al., 2013, 2014, 2012]. Finally, after completing all previous steps, all tweets were indexed with the use of the Apache Lucene library<sup>4</sup>.

## 6.2 Experiment Measures

Evaluation is crucial for the design, develop and maintenance of efficient IR models as it enables the measurement of how well an IR system meets its objective to help users fulfil their needs. Fulfilment is typically characterised by the number of relevant results the system gives and whether those results are ranked. Different measurements characterised retrieval effectiveness for a given user's information need. The most basic group of retrieval measurements are set-based, such as precision and recall. Different means, based on the precision and recall measures, will be used in this research, including Mean Average precision (MAP), the precision at a specific cut-off, the normalised discounted cumulative gain (NDCG), and R-precision (Rprec).

In this thesis, we utilised a set of relevance oriented measures, including P@k, MAP, NDCG, and Rprec. One of the reasons that these evaluation measures are used is to compare the proposed models with previous models. Following the TREC microblog guidelines [Ounis et al., 2011, Soboroff et al., 2013, 2014, 2012], this research used the official P@30 measure and considered both minimally and highly relevant tweets. Different evaluation measures also are reported, including P@10,

---

<sup>4</sup><http://lucene.apache.org/>

**Table 6.2:** Contingency Table

	Relevant	Not Relevant
Retrieved	True Positive (TP)	False Positive (FP)
Not Retrieved	False Negative (FN)	True Negative (TN)

MAP, and Rprec. Since some tweets are identified as relevant or highly relevant to queries, NDCG [Järvelin and Kekäläinen, 2002] considers assessing the quality performance of the proposed as well as the baselines models if appropriate. We utilise the TREC\_eval<sup>5</sup> to compute the performance scores. These measures are extensively used in microblog retrieval research.

Precision and recall are both common and fundamental metrics for information retrieval effectiveness. These measures are defined when an IR system returns a set of documents for a given user query. As in contingency table 6.2, precision and recall can be defined, as follows:

- **Precision** ( $p$ ): For a given topic or query, precision represents the ratio between the total number of relevant retrieved documents to all the retrieved documents. Precision can be computed by the following formula:

$$Precision(p) = \frac{|\{\#relevant\ documents\} \cap \{\#retrieved\ documents\}|}{|\{\#retrieved\ documents\}|} = \frac{TP}{TP + FP}$$

- **Recall** ( $r$ ): For a given topic or query, recall represents the ratio between the total number of relevant retrieved documents to all the relevant documents. Recall can be computed by the following formula:

<sup>5</sup>The evaluation tool can be downloaded from [https://trec.nist.gov/trec\\_eval/](https://trec.nist.gov/trec_eval/).



$$Recall(r) = \frac{|\{\#relevant\ documents\} \cap \{\#retrieved\ documents\}|}{|\{\#relevant\ documents\}|} = \frac{TP}{TP + FN}$$

The terminology defines the judgments as represented in contingency table 6.2. TP (True Positive) refers to the number of relevant tweets that the system retrieved, while FP (False Positive) is the number of relevant tweets that the system could not retrieve. FN (False Negative) is the number of tweets the system does not identify, and TN (True Negative) is the number of tweets the system correctly identifies as irrelevant.

- **Mean Average Precision (MAP)** is the computed average precision over all topics or queries. Specifically, the average precision for a single query or topic is the precision value that is calculated for a set of top  $k$  document.

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \quad (6.1)$$

- **Normalized Discounted Cumulative Gain (NDCG)**: The main objective of normalised discounted cumulative gain (NDCG) is to evaluate with multi-level judgments. The top relevant results when using these type of judgments are called “gains”. Gains commonly matches the utility of a document from a user’s needs. To figure out position-based penalty, the discounted cumulative gain (DCG) can be defined as follows:

$$DCG(L) = rel_1 + \sum_{i=2}^n \frac{rel_i}{\log_2 i} \quad (6.2)$$

Each document's gain in the top  $n$  ranking list is discounted by dividing by a logarithm of its position in the list Zhai and Massung [2016]. The normalised discounted cumulative gain (NDCG) is then a discounted cumulative gain (DCG) regarding an ideal discounted cumulative gain ranking (IDCG). Finally, the normalised discounted cumulative gain is comparable across different queries and computes as follow:

$$NDCG(L) = \frac{DCG(L)}{IDCG} \quad (6.3)$$

Multi-level relevance judgment is a challenging evaluation task. Thus, a significant metric dealing with measuring multi-relevance judgment is NDCG. Generally, NDCG can be applied through any ranked application with a multi-level relevance judgment. The primary aim of this measure is to outline the aggregate utility of the top  $k$  documents. Lastly, it performs normalisation to guarantee likeness crosswise over queries or topics.

- **R-precision** (Rprec) is the ratio of the top retrieved document that are relevant. In other words, it is precision at  $R$  where  $R$  is the number of relevant documents in a collection for a given topic. Rprec is defined as  $\frac{r}{R}$ , where  $r$  is the number of relevant documents that are retrieved in the ranked list using the proposed algorithm. For example, if we assume that there are 100 documents in a collection and 30 documents are relevant where the rest is not relevant. If the ranked list only includes 10 relevant out of 30, thus, Rprec is  $\frac{10}{30} = \frac{1}{3}$ . The main benefit of Rprec measure evaluation is that it has lower error rates compared with Precision [Sanderson and Zobel, 2005].

Moreover, the statistical significance of results obtained by the proposed models were tested using the two-tailed paired  $t$ -test with the  $p$  value to validate the retrieval

effectiveness. Smucker et al. [2007] demonstrates that there is no distinction between  $t$ -test and randomisation test in practice, where the latter is a more basic option. In this thesis, we select the  $t$ -test because we need to promote retrieval models that are better than other models which have, by chance, worked better given the various topics, judgments, and documents used in the assessment.

### 6.3 Baseline Models

The proposed models will be compared to other models, including state-of-the-art models. In the experiments, we compare our proposed models (TAPRF, TBS and QUSTM) with probabilistic model (BM25), language model (QL), temporal language model (Recency), a state-of-the-art temporal feedback method (KDE), a state-of-the-art pseudo relevance feedback model (RM3), a topic model (LDA), and a state-of-the-art short text-based topic model (PTM). These baselines algorithms are described as follows:

- The first baseline is a probabilistic state-of-the-art retrieval model (BM25) Robertson et al. [1994] that that can compute the similarity between document  $d$  and query  $Q$  containing words  $w$  as the following equation:

$$BM25(Q, d) = \sum_{w \in Q \cap d} IDF(w) \times \frac{(k_1 + 1)c(w, d)}{k_1((1 - b) + b \frac{dl}{avdl}) + c(w, d)}$$

where  $IDF(w)$  is estimate, as follows:

$$IDF(w) = \log \frac{N - df(d) + 0.5}{df(d) + 0.5}$$

- In the second baseline, the query likelihood model with Dirichlet smoothing

(referred as QL) is utilising the Dirichlet smoothing parameter  $\mu = 100$  based on settings in paper Lv et al. [2015].

- Recency as one of the robust time-based retrieval model to let time influence the ranking model was given by Li and Croft [2003], who proposed a document prior that favors recently published documents. If  $T_d$  is the timestamp associated with document  $d$ , they propose modeling  $P(d)$  in  $P(d|Q) \propto P(Q|d)P(d)$  via an exponential distribution  $P(d) = \lambda e^{-\lambda T_d}$ , where  $\lambda \geq 0$  is the proportion parameter of the exponential distribution.
- Kernel Density Estimation (KDE) is a state-of-the-art model that estimates the temporal density of relevance feedback for microblog documents [Efron et al., 2014]. KDE is a non-parametric approach to estimating the probability density function of the distribution from the observations. KDE attempts to place a kernel on each point and then sums them up to discover the overall distribution. Its kernel density estimator is as follows:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=0}^n K\left(\frac{x - x_i}{h}\right)$$

where  $K(\cdot)$  is the kernel, a symmetric but not necessarily positive function that integrates to one, and  $h > 0$  is a smoothing parameter called the bandwidth. Though many kernel functions are viable, we followed the paper and used the common Gaussian distribution, as follows:

$$K\left(\frac{x - y}{h}\right) = \mathcal{N}\left(\frac{x - y}{h}, 0, h\right)$$

where  $\mathcal{N}$  is the normal density. A kernel density estimate is very similar to

a histogram. One key advantage of using KDE versus histograms for estimating  $f$  is KDE's ability to handle weighted observations naturally. If we have  $\{\omega_1, \omega_2, \dots, \omega_n\}$ , a vector of non-negative weights on our observed  $X$ 's such that  $\sum \omega_i = 1$ .

- A PRF relevance model RM3 model [Abdul-Jaleel et al., 2004] is used to compare with our proposed models. In RM3, for a given query, the relevance model is as below and then interpolated with the original query with a control parameter.

$$P(w|R) = \sum_{d \in F} P(d)P(w|d) \prod_n^{i=1} P(q_i|d)$$

where the relevance model  $P(w|R)$  is as estimate of the pseudo relevance feedback, and  $F$  is the number of top pseudo feedback in this paper ( $F = 50$ ). Then, the relevance feedback is interpolated with the original query model as follows:

$$P(w|Q') = \kappa P(w|Q) + (1 - \kappa) P(w|R)$$

where  $\kappa \in [0, 1]$  is a parameter to balance the using of pseudo-relevance feedback, and, in this thesis, we stted  $\kappa$  as our  $\lambda$ .

- LDA is a state-of-the-art topic model that finds the latent topics for a given collection [Blei et al., 2003]. We described LDA in Section 3.2.1.
- Pseudo document-based topic modeling (PTM) is an innovative topic modeling approach that is designed for short-text analysis [Zuo et al., 2016]. We exploited this as the proposed model in order to be fair; thus, the input for this model will be the number of top-ranked tweets. For the parameter settings for this baseline, we followed the paper [Shi et al., 2018].

## 6.4 Experiment Settings

All the experiments stated in this thesis have been executed on a PC with an Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz and 16 GB memory running a Windows 7 operating system. The system of the research project was programmed using Java programming language with J2SDK version 1.8.0 as the development environment.

The Dirichlet prior smoothing parameter  $\mu$  in QL with Dirichlet prior swept over values from 50 to 1000 at an interval of 50. In the meantime, we sweep from 0 to 1.0 the  $b$  values for BM25 with an interval 0.1. The number of tweets and the terms selected are set using two-fold cross-validation over each collection. We swept the tweets' feedback between 10 to 100 with an interval of 10 and then selected number terms between 10 to 100 with an interval of 5. The parameters that are used in the baseline models, if required, are also set by utilising the same process.

Different experimental parameter settings were used within our framework. In our experimental framework, we utilised the Java Machine Learning for Language Toolkit (MALLET)<sup>6</sup> in our experimental environment system. The hyperparameters settings for the LDA model was  $\alpha = 50/V$  and  $\beta = 0.01$  as was recommended in Chuang et al. [2013].

## 6.5 TAPRF Evaluation

### 6.5.1 Overall Results

To verify the first hypothesis, we compare the proposed model's (TAPRF) performance with the baseline models. The results obtained by the proposed model and the baselines

---

<sup>6</sup><http://mallet.cs.umass.edu/>

**Table 6.3:** Comparison of the proposed method TAPRF and baselines lexical based models.

MB2011					
Model	P.10	P.30	MAP	NDCG	Rprec
BM25	0.4388	0.3599	0.3310	0.5715	0.3848
QL	0.4592	0.3714	0.3561	0.5940	0.3993
TAPRF	<b>0.4714</b>	<b>0.4020<sup>1,2</sup></b>	<b>0.3957<sup>1,2</sup></b>	<b>0.6198<sup>1,2</sup></b>	<b>0.4234<sup>1,2</sup></b>
<i>ch%</i>	+2.66%	+8.24%	+11.12%	+4.34%	+6.04%
MB2012					
BM25	0.3915	0.3270	0.2118	0.4690	0.2691
QL	0.4017	0.3327	0.2248	0.4822	0.2823
TAPRF	<b>0.4492<sup>1,2</sup></b>	<b>0.3638<sup>1,2</sup></b>	<b>0.2651<sup>1,2</sup></b>	<b>0.5242<sup>1,2</sup></b>	<b>0.3154<sup>1,2</sup></b>
<i>ch%</i>	+11.82%	+9.35%	+17.93%	+8.71%	+11.73%
MB2013					
BM25	0.5850	0.4383	0.2603	0.4759	0.3038
QL	<b>0.6050</b>	0.4544	0.2825	0.4945	0.3250
TAPRF	0.5867	<b>0.4839<sup>2</sup></b>	<b>0.3193<sup>1,2</sup></b>	<b>0.5473<sup>1,2</sup></b>	<b>0.3488<sup>2</sup></b>
<i>ch%</i>	-3.02%	+6.49%	+13.03%	+10.68%	+7.32%
MB2014					
BM25	0.7418	0.6345	0.4300	0.6503	0.4477
QL	0.7364	0.6558	0.4573	0.6711	0.4757
TAPRF	<b>0.7818</b>	<b>0.6836<sup>2</sup></b>	<b>0.5302<sup>1,2</sup></b>	<b>0.7263<sup>1,2</sup></b>	<b>0.5154<sup>1,2</sup></b>
<i>ch%</i>	+6.17%	+4.24%	+15.94%	+8.23%	+8.35%

The highest value in each test set is marked in bold; superscripts 1, 2 indicate statistically significant improvement at ( $p < 0.05$ ) over QL and BM25; and the *ch%* column denotes the improvements over QL.

are presented in Table 6.3. As in Table 6.3, it reports the overall TAPRF experimental results for the TREC microblog datasets 2011-2014 based on well-known evaluation metrics that include precision at a rank 10 and 30, respectively; Mean Average Precision (MAP), Normalised Document Gain Cumulative (NDCG); and R-Precision (Rprec) where *%chg* presents the percentage change of TAPRF over QL and the best results are emphasised in bold. The lexical-based retrieval models (including QL and BM25) provided a baseline for the proposed model's performance results. The statistical significance tests, where all  $p$  values are less than 0.05, are marked in the upper right-hand corner of TAPRF's performance scores.

According to Table 6.3, the QL model baseline approach slightly outperforms the probabilistic BM25 model in most cases. The proposed model (TAPRF) outperforms all the baselines based on of P.10, P.30, MAP, NDCG and Rprec in all query test sets over Tweets2011 and Tweets2013 collections. The statistical  $t$ -test shows has significant improvements for TAPRF based on most evaluation metrics over baselines models in all query test sets. These improvements over the lexically based baselines are also always significant. These results show the effectiveness of the proposed method compared to state-of-the-art baselines approaches.

It can be clearly seen experimentally that TAPRF outperforms QL and BM25 for TREC 2011 and 2012. Table 6.3 shows that TAPRF achieves excellent performance with a 6.48 percent improvement, on average, for test set MB2011 on Tweets2011 (with a maximum of 11.12 percent and a minimum of 2.66 percent). For test set MB2012 on Tweets2011, TAPRF reaches outstanding performance, with 11.91 percent improvement on average (with a maximum of 17.93 percent and a minimum of 8.71 percent). For test set MB2013 on Tweets2013, TAPRF has achieved an average of 6.90 percent (maximum 13.03 percent and a minimum of -3.02 percent). TAPRF achieves exceptional performance with 8.59% on average (maximum 15.94 percent



**Table 6.4:** Example of expanded terms for a topic numbered MB86: “Michelle Obama’s obesity campaign”.

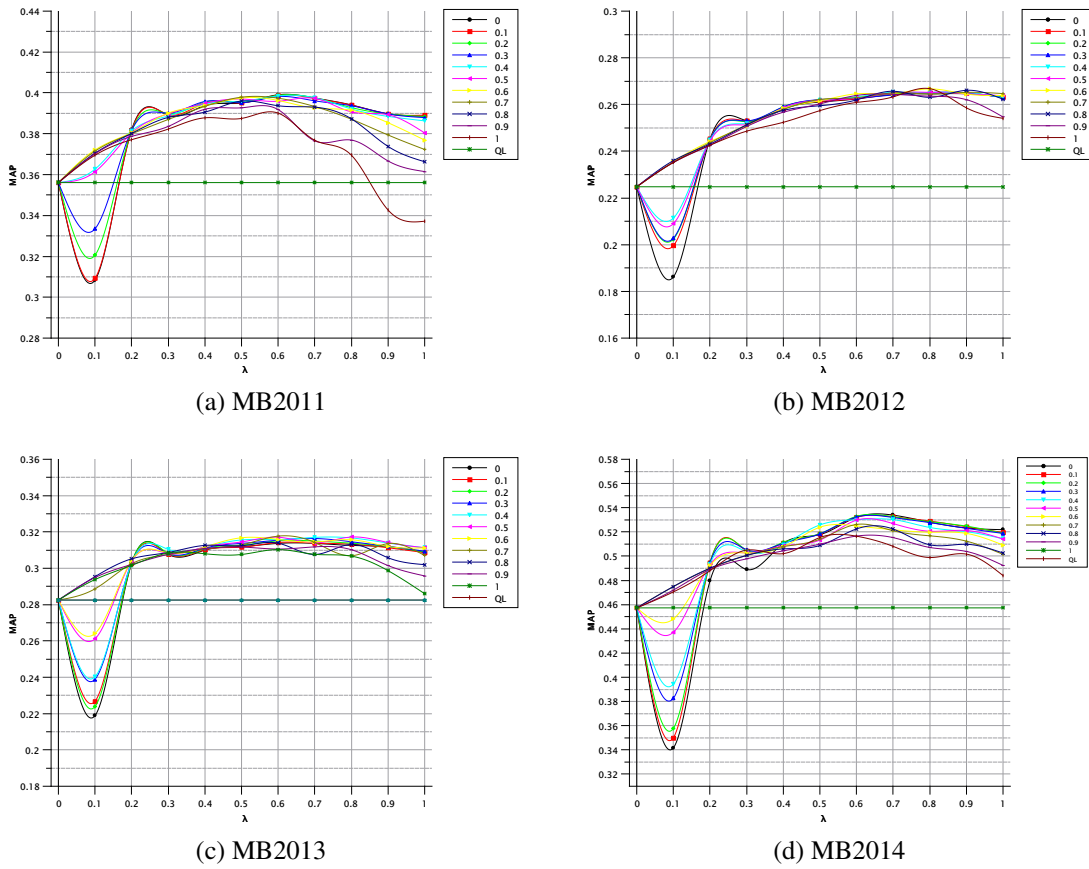
$P_{exp}(w Q)$		$P'(w Q)$	
term	weight	term	weight
obama	0.2487	obama	0.2527
michelle	0.2284	micHEL	0.2382
campaign	0.1117	campaign	0.1863
lady	0.0609	obes	0.1472
obesity	0.0457	ladi	0.0331
oprah	0.0355	atlanta	0.0157
atlanta	0.0254	childhood	0.0122
obama’s	0.0254	move	0.0104
stop	0.0254	oprah	0.0092
childhood	0.0203	stop	0.0091
food	0.0203	militari	0.0083
coming	0.0203	role	0.0080
weight	0.0203	plai	0.0079
role	0.0203	utm	0.0076

and minimal 4.24 percent) improvements for test set MB2014 on Tweets2013.

Table 6.4 shows an example of query analysis for topic number MB86 with the topic title “Michelle Obama’s obesity campaign”. It presents the difference between the relevance using the relevance model and our TAPRF relevance model estimation for nominated terms for expansion.

### 6.5.2 Discussion

In this section, we discuss the results for the proposed model regarding parameters tuning, feedbacks and topic number sensitivity, as well as per-query analysis across all test sets.



**Figure 6.4:** TAPRF parameters sensitivity.

### 6.5.2.1 Parameters tuning

Several parameters in the proposed model can affect the retrieval system's performance. This section analyses the robustness to the parameter settings in interpolation between the lexical and topical evidence as in Equation 3.10 and the smoothing parameter as in Equation 3.11 that integrates the original query model with the proposed weighting schema.

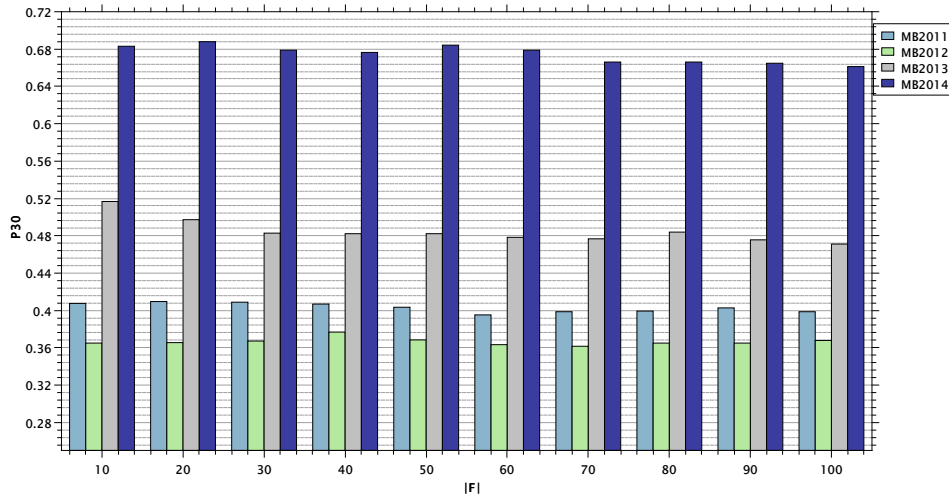
There are two main parameters,  $\lambda$  and  $\gamma$ , in the proposed model to balance between the lexical and topical evidence. Specifically, when  $\lambda = 0$  ignore the topical evidence

and only use the relevance model; where  $\lambda = 1$  ignore the relevance model and use the topical evidence. When  $\gamma = 1$  means ignoring the score that comes from Equation 3.11 and only considering the original query model, but when  $\gamma = 0$  refer to view the weight from the previous Equation 3.11. Both parameters tested in range 0 to 1 overall test sets, as shown in Figure 6.4. Figure 6.4 includes four subfigures where each figure showed TAPRF performance against both parameters. The legend in these figures represents the change of  $\gamma$  with different values, and the y-axis represents the  $\lambda$  value. We only showed the MAP to see how the change of parameters values the evidence effectiveness as proposed.

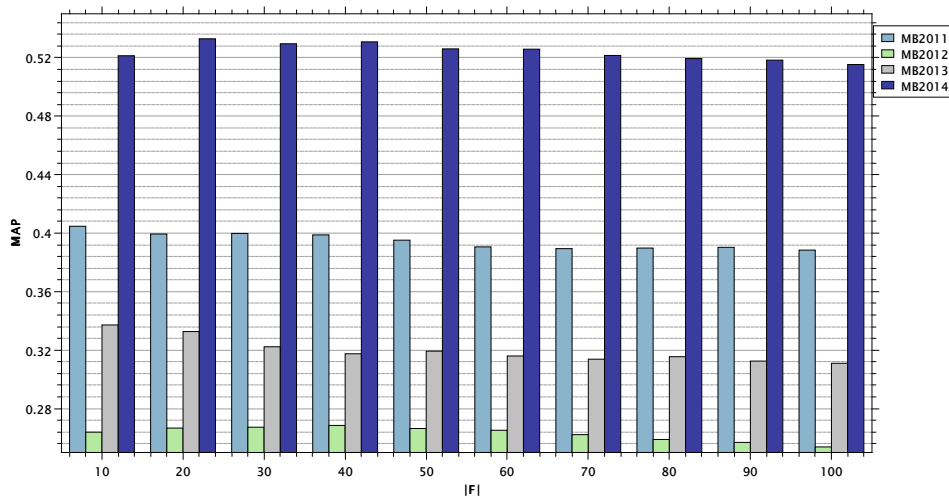
From Figure 6.4, it is worthwhile to point out that TAPRF performance is effective when the  $\lambda > 0.6$  for all test sets. A rational reason for this phenomenon is that the initial pseudo results set tends to include topical evidence more than that in the lexical evidence. As we mentioned earlier,  $\lambda = 1$  means to only consider the topical evidence; therefore, the result is best than the lexical evidence only. However, TAPRF assumes the relevant information is feeding from lexical and topical evidence; therefore, TAPRF achieved optimal performance when the  $\lambda$  was between 0.5 to 0.8 and the  $\gamma$  showed less sensitivity across all the ranges.

### 6.5.2.2 Feedbacks Sensitivity

From Figure 6.5, we present the proposed model TAPRF's performance on P30 and MAP metrics against a different number of pseudo-documents  $F$  (from 10 to 100) from overall test sets MB2011, MB2012, MB2013 and MB2014. For the P30 metric, it clearly showed when the pseudo-documents set is  $|F| > 50$ , TAPRF showed the best performance across test sets. For the MAP metric, TAPRF performance slightly decreased when the pseudo-documents set  $|F|$  greater than 20 tweets.



(a) P30

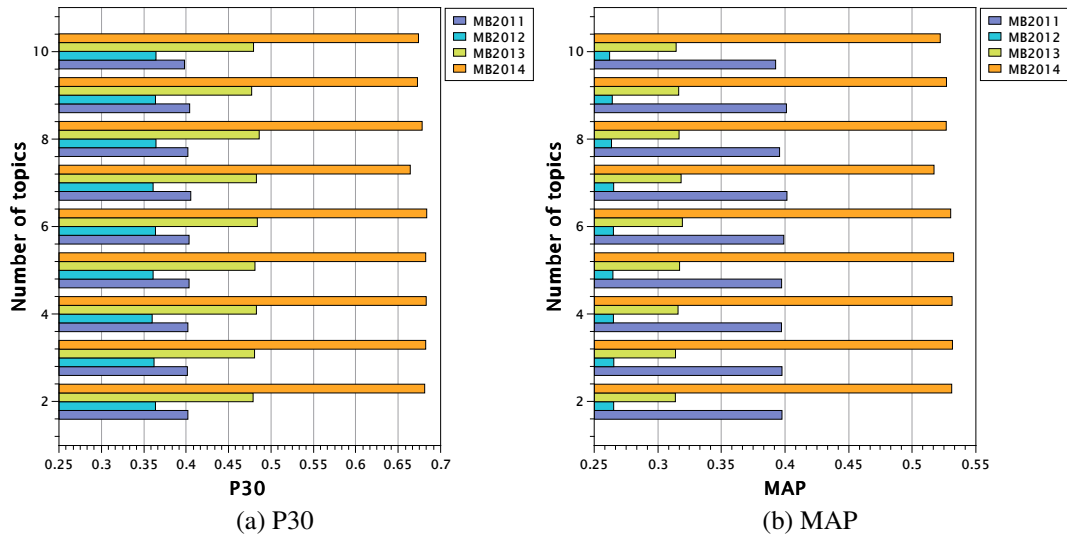


(b) MAP

**Figure 6.5:** TAPRF number of pseudo-feedbacks sensitivity.

### 6.5.2.3 Topics Sensitivity

The proposed TAPRF model used the topic model (LDA) as a implies to discover the latent topics, as discussed in Section 3.3.1. For the proposed model TAPRF, we analysed the performance change in the number of topics used in the LDA based on the TREC official metrics P30 and MAP overall test sets. Figure 6.6.(a) illustrates TAPRF



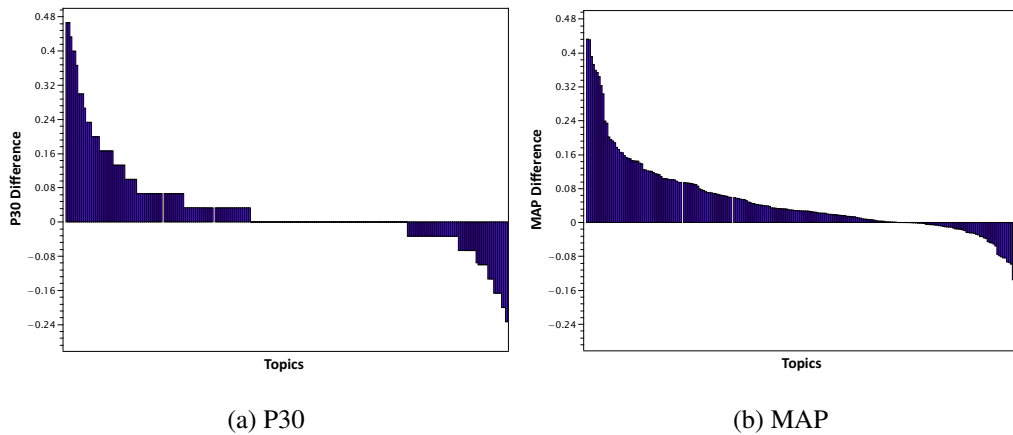
**Figure 6.6:** TAPRF number of topics sensitivity.

performance based on P30 value and (b) based on MAP value in a number of topics (2 to 10).

For the MB2011 test set, TAPRF achieved the optimal results based on P30 and MAP when 7 or 9 topics were selected. For both metrics in the MB2012 test set, TAPRF performance was stable across all tested topic numbers. Moreover, the proposed TAPRF model showed strength against the change in the number of topics for both evaluation metrics in the MB2013 test set. Finally, for the MB2014 test set against P30 and MAP, the proposed model TAPRF performance was stable compared to the number of topics.

#### 6.5.2.4 Per-Query Analysis

The P30 differential between TAPRF and QL shows in Figure 6.7.(a) on the basis of the query. The proposed TAPRF model improved results for approximately 92 queries over 223 across test sets while in nearly 51 queries lowered effect; however, Figure



**Figure 6.7:** TAPRF per-query analysis.

6.7.(b) shows, on a subject-based basis, the difference in the P30 results for TAPRF. TAPRF improves retrieval performance in test sets for most queries, where results for around 158 queries have improved, and outcomes for about 60 queries have decreased.

## 6.6 TBS Evaluation

### 6.6.1 Overall Results

Table 6.5 and Table 6.6 show the proposed model TBS’s performance compared with the baseline models regarding P@10, P@30, MAP, and NDCG evaluation metrics across all test sets. Both tables presented *ch%*, which refers to the change in percentage and the statistically significant improvements over the state-of-the-art baseline model RM3, where all *p* values were less than 0.05. In each evaluation metric column, the best result is marked in bold font. All the parameters were tuned on the Tweets2011 dataset with MB2011 topics. Then, we tested the proposed model TBS’s performance on the Tweets2011 and Tweets2013 datasets with MB2012, MB2013, and MB2014 topic sets.

**Table 6.5:** Comparison of the proposed method TBS and baselines models over MB2011 and MB2012 test sets.

MB2011					
Model	P.10	P.30	MAP	NDCG	Rprec
BM25	0.4388	0.3599	0.3310	0.5715	0.3848
QL	0.4592	0.3714	0.3561	0.5940	0.3993
Recency	0.4673	0.3776	0.3581	0.5956	0.4019
KDE	0.4633	0.3741	0.3398	0.5782	0.3554
RM3	0.4714	0.3986	0.3712	0.5819	0.3986
TBS	<b>0.4980</b>	<b>0.4204</b> <sup>1,2</sup>	<b>0.4249</b> <sup>1,2,3</sup>	<b>0.6530</b> <sup>1</sup>	<b>0.4578</b> <sup>1,2,3</sup>
<i>ch%</i>	+8.45%	+13.19%	+19.32%	+9.93%	+14.65%
MB2012					
BM25	0.3915	0.3270	0.2118	0.4690	0.2691
QL	0.4017	0.3327	0.2248	0.4822	0.2823
Recency	0.4051	0.3349	0.2255	0.4824	0.2830
KDE	0.3898	0.3316	0.2249	0.4778	0.2855
RM3	0.4136	0.3627	0.2534	0.5004	0.3047
TBS	<b>0.4169</b> <sup>1</sup>	<b>0.3757</b> <sup>1,2</sup>	<b>0.2691</b> <sup>1,2</sup>	<b>0.5506</b> <sup>1,2,3</sup>	<b>0.3256</b> <sup>1,2,3</sup>
<i>ch%</i>	+3.78%	+12.92%	+19.71%	+14.18%	+15.34%

The highest value in each test set is marked in bold; superscripts 1,2 and 3 indicate statistically significant improvement at ( $p < 0.05$ ) over QL, KDE and RM3; and the *ch%* column denotes the improvements over QL.

As the results in Table 6.5 show, the proposed model (TBS), overall outperformed the baseline models across all test sets regarding the evaluation metrics. Specifically, from Table 6.5 for the MB2011 topic set, TBS improved the MAP by a maximum of 28.37% compared to BM25 and a 14.47% minimum against compared to RM3. Also, TBS improved the P.30 by a maximum of 16.81% compared to BM25 and a 5.47%

**Table 6.6:** Comparison of the proposed method TBS and baselines models over MB2013 and MB2014 test sets.

MB2013					
Model	P.10	P.30	MAP	NDCG	Rprec
BM25	0.5850	0.4383	0.2603	0.4759	0.3038
QL	0.6050	0.4544	0.2825	0.4945	0.3250
Recency	0.6100	0.4694	0.2875	0.4993	0.3330
KDE	0.6117	0.4644	0.2791	0.4917	0.3273
RM3	0.5150	0.4467	0.3035	0.5102	0.3310
TBS	<b>0.6400</b> <sup>3</sup>	<b>0.5228</b> <sup>1,2,3</sup>	<b>0.3513</b> <sup>1,2,3</sup>	<b>0.5936</b> <sup>1,2,3</sup>	<b>0.3830</b> <sup>1,2,3</sup>
<i>ch%</i>	+5.79%	+15.05%	+24.35%	+20.04%	+17.85%
MB2014					
BM25	0.7418	0.6345	0.4300	0.6503	0.4477
QL	0.7364	0.6558	0.4573	0.6711	0.4757
Recency	0.7436	0.6552	0.4606	0.6742	0.4767
KDE	0.7218	0.6539	0.4641	0.6700	0.4840
RM3	0.7436	0.6467	0.4951	0.6945	0.4785
TBS	<b>0.7618</b> <sup>1</sup>	<b>0.6903</b> <sup>1</sup>	<b>0.5189</b> <sup>1</sup>	<b>0.7248</b> <sup>1</sup>	<b>0.5147</b> <sup>1</sup>
<i>ch%</i>	+3.45%	+5.26%	+13.47%	+8.00%	+8.20%

The highest value in each test set is marked in bold; superscripts 1,2 and 3 indicate statistically significant improvement at ( $p < 0.05$ ) over QL, KDE and RM3; and the *ch%* column denotes the improvements over QL.

minimum compared to RM3. For the MB2012 topic set, TBS improved over the MAP by a maximum of 27.05% compared to BM25 and a minimum of 6.20% compared to RM3. Furthermore, TBS improved over the P.30 by a maximum improvement of 14.89% compared to BM25 and enhanced by a minimum of 3.58% compared to RM3.

To verify the superiority of the proposed model, we evaluated TBS on another two



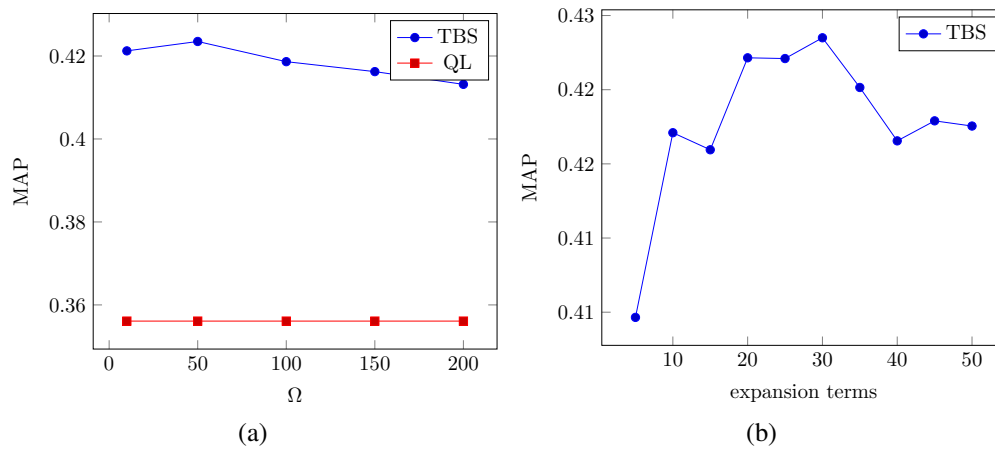
test sets over the Tweets2013 dataset, which was much larger than the Tweets2011 dataset, and presented the diversity in the proposed model performance. From Table 6.6, for the MB2013 test set, the proposed model TBS the corresponding increases of P.30 were a maximum of 19.28% and a minimum of 11.38% over BM25 and Recency, respectively, while improved the MAP by a maximum and minimum of 34.96% and 15.51% over BM25 and RM3, respectively. For the MB2014 test set, TBS increased the P.30 by a maximum of 8.79% compared to BM25 and a 5.26% minimum compared to QL. The proposed model TBS improved the MAP by a maximum of 20.67% compared to BM25 and a 4.81% minimum improvement compared to RM3. This result demonstrates that the proposed model is a valid hypothesis for social media search tasks.

### 6.6.2 Discussion

In this section, we provide a discussion of the proposed model's results. First, we cover a parameter sensitivity analysis of the model. All analyses of parameter settings were compared on the MB2011 topic set that was utilised for parameter tuning over the Tweets2011 dataset. Second, we examined the per-query improvements for all test sets of the proposed model.

#### 6.6.2.1 Effects for number of feedback documents

In Figure 6.8.(a), we presents the proposed model TBS's performance on MAP value against a different number of top-ranked documents  $|\Omega|$  (from 10 to 200). As is clearly seen, the proposed model TBS obtained the optimal performance when  $\Omega = 50$ . In addition, the performance of the proposed model TBS slightly decreased when  $|\Omega| > 50$ . Based on this observation, we fixed the number of top-ranked documents  $\Omega$  to 50.



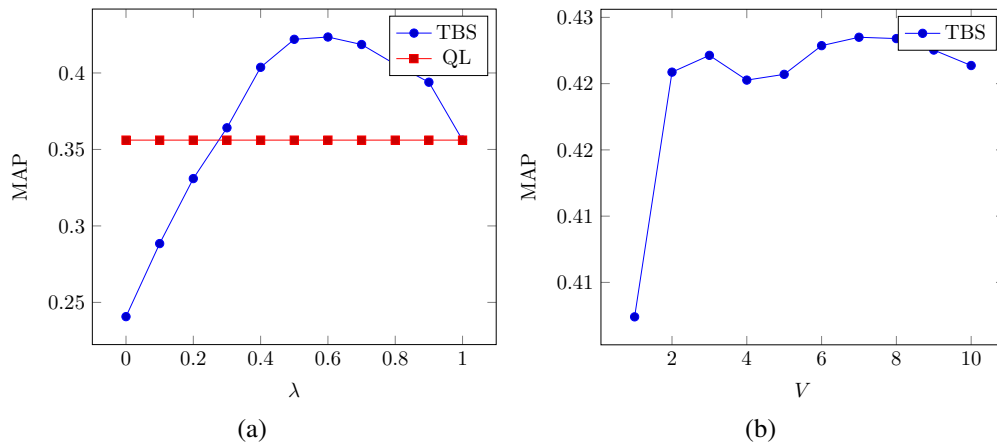
**Figure 6.8:** Sensitivity of the number of feedback documents  $\Omega$  in (a) and the number of expansion terms in (b) on TBS for the microblog TREC 2011 collection.

### 6.6.2.2 Effects for number of expansion terms

Figure 6.8.(b) shows the proposed model TBS's performance sensitivity on MAP value compared to the number of expansion features. The proposed model TBS obtained the optimal performance when the number of expansion terms was even to 30. The variation of the proposed model TBS's performance regarding the change of expansion features is a slight change with other numbers of expansion terms. Thus, the top 30 expansion features can provide enough information for retrieving good relevant information about the original user information needs.

### 6.6.2.3 Effects of the interpolation parameter

In Figure 6.9.(a), we visualise the proposed model TBS's performance on MAP value against different feedback coefficients  $\lambda$ . In Section 4.2.2, we discussed the integration between the topical information and the relevance feedback model and then interpolated the original query model, as in Equation 4.3 and 4.5. When the feedback

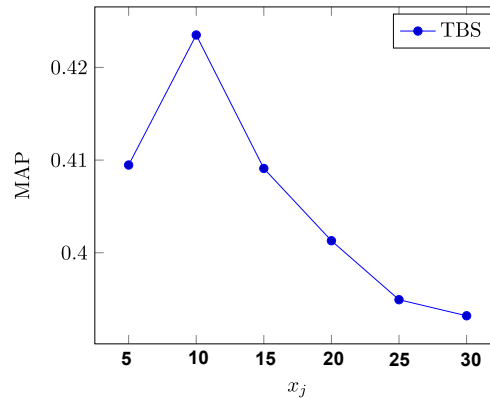


**Figure 6.9:** Sensitivity of the number of interpolation parameters  $\lambda$  in (a) and the number of topics  $V$  in (b) on TBS for the microblog TREC 2011 collection.

coefficient  $\lambda$  equal 1, we completely ignored using the relevance feedback model and the proposed model TBS performance decreases into the input model QL. Where the feedback coefficient  $\lambda$  equals 0, we only used the relevance feedback without the original query. It is clearly seen from Figure 6.9.(a) that the proposed model TBS obtained the significant performance when the feedback coefficient  $\lambda$  equal 0.6.

#### 6.6.2.4 Effects of the topics number

As we discussed in Section 4.2.2, the proposed model TBS utilised the topic model (LDA) to discover the latent topics. We analysed the change in the number of the LDA topics for the proposed model TBS's performance. Figure 6.9.(b) illustrates the proposed model's performance on MAP value across a different number of topics (from one to ten). The proposed model TBS's performance became more stable when the number of topics  $V$  equald one. The proposed model TBS's performance showed robustness against the change in the number of topics, especially from two to ten. The proposed model TBS obtained the best performance when the number of topics equals



**Figure 6.10:** Sensitivity of the sequence value  $x_j$  on TBS at microblog TREC 2011 collection.

seven.

#### 6.6.2.5 Effects of the subsequence value

As discussed in Section 4.2.1, the proposed model TBS's performance can be affected by the subsequence value  $x_j$ . From Figure 6.10, we demonstrate the proposed model TBS's performance on MAP value against the change of the subsequence value  $x_j$ . The proposed model TBS achieved the best performance when an incremental parameter  $x_2$  value equals 10.

#### 6.6.2.6 Per-query analysis

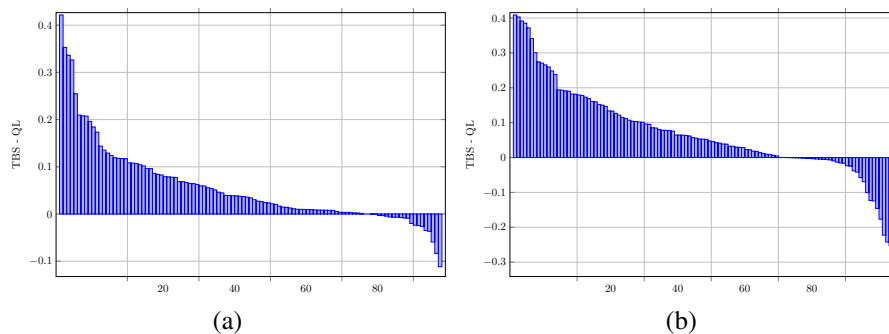
Figure 6.11.(a) shows the difference in MAP between TBS and QL on a topic-by-topic basis. The proposed model TBS improved results for approximately 88 topics and decreased results for approximately 20 topics using MB2011 and MB2012. Furthermore, Figure 6.11.(b) shows the results for TBS on a topic-by-topic basis. TBS improves the retrieval performance for most topics in MB2013 and MB2014. The proposed

TBS model improved results for approximately 82 topics and decreased results for approximately 33 topics.

## 6.7 QUSTM Evaluation

### 6.7.1 Overall Results

To verify the third hypothesis, we compare the proposed (QUSTM) model's performance with the baseline models in the TREC microblog datasets 2011-2014. The results obtained by the proposed model and the baselines are presented in Table 6.7 and 6.8. According to these tables, the temporal baseline approaches slightly outperform QL in most cases, as well as the topic modeling based approaches. In Table 6.7 and 6.8, the proposed model outperforms all the baselines based on P.30 and MAP that were the official metrics of the TREC microblog in all query test sets over Tweets2011 and Tweets2013 collections. The statistical  $t$ -test shows that the P.30 and MAP improvements over QL are significant in all query test sets. These improvements over the strong baselines are also always significant. These results show the effectiveness of the proposed method compared to state-of-the-art baseline approaches.



**Figure 6.11:** Difference in MAP between TBS and QL using the MB2011-MB2012 topic sets in (a) and the MB2013 and MB2014 topic sets in (b).

**Table 6.7:** Comparison of the proposed method QUSTM and baselines models over MB2011 and MB2012 test sets.

MB2011					
Model	P.10	P.30	MAP	NDCG	Rprec
BM25	0.4388	0.3599	0.3310	0.5715	0.3848
QL	0.4592	0.3714	0.3561	0.5940	0.3993
Recency	0.4673	0.3776	0.3581	0.5956	0.4019
KDE	0.4490	0.3735	0.3338	0.5734	0.3554
RM3	0.4714	0.3986	0.3712	0.5819	0.3986
LDA	0.4299	0.3902	0.3347	0.5698	0.4145
PTM	0.4694	0.3966	0.3506	0.5558	0.3597
QUSTM	<b>0.4816</b> <sup>1,4</sup>	<b>0.4102</b> <sup>1,2,3</sup>	<b>0.4127</b> <sup>1,2,3,4</sup>	<b>0.6230</b> <sup>1,2,3,4</sup>	<b>0.4309</b> <sup>1,2,3</sup>
<i>ch%</i>	+4.88%	+10.45%	+15.89%	+4.88%	+7.91%
MB2012					
BM25	0.3915	0.3270	0.2118	0.4690	0.2691
QL	0.4017	0.3327	0.2248	0.4822	0.2823
Recency	0.4051	0.3349	0.2255	0.4824	0.2830
KDE	0.3898	0.3316	0.2249	0.4778	0.2855
RM3	0.4136	0.3627	0.2534	0.5004	0.3047
LDA	0.3932	0.3586	0.2353	0.4883	0.2935
PTM	0.4203	0.3429	0.2466	0.4870	0.3280
QUSTM	<b>0.4559</b> <sup>1,2,3,4</sup>	<b>0.3876</b> <sup>1,2,3,4</sup>	<b>0.2814</b> <sup>1,2,3,4</sup>	<b>0.5421</b> <sup>1,2,3,4</sup>	<b>0.3280</b> <sup>1,2,3,4</sup>
<i>ch%</i>	+13.49%	+16.50%	+25.18%	+12.42%	+16.19%

The highest value in each test set is marked in bold; superscripts 1,2,3 and 4 indicate statistically significant improvement at ( $p < 0.05$ ) over BM25, QL, KDE and LDA; and the *ch%* column denotes the improvements over QL.

It can be clearly seen in the experiment that the proposed model outperformed and showed significant improvement over the baseline models in all metrics across

**Table 6.8:** Comparison of the proposed method QUSTM and baselines models over MB2013 and MB2014 test sets.

MB2013					
Model	P.10	P.30	MAP	NDCG	Rprec
BM25	0.5850	0.4383	0.2603	0.4759	0.3038
QL	0.6050	0.4544	0.2825	0.4945	0.3250
Recency	0.6100	0.4694	0.2875	0.4993	0.3330
KDE	0.6117	0.4644	0.2791	0.4917	0.3273
RM3	0.5150	0.4467	0.3035	0.5102	0.3310
LDA	0.4750	0.4504	0.2755	0.4751	0.3351
PTM	0.5367	0.4578	0.2864	0.4859	0.3105
QUSTM	<b>0.6650</b> <sup>1,2,3,4</sup>	<b>0.5428</b> <sup>1,2,3,4</sup>	<b>0.3693</b> <sup>1,2,3,4</sup>	<b>0.5919</b> <sup>1,2,3,4</sup>	<b>0.3974</b> <sup>1,2,3,4</sup>
<i>ch%</i>	+9.92%	+19.45%	+30.73%	+19.70%	+22.28%
MB2014					
BM25	0.7418	0.6345	0.4300	0.6503	0.4477
QL	0.7364	0.6558	0.4573	0.6711	0.4757
Recency	0.7436	0.6552	0.4606	0.6742	0.4767
KDE	0.7218	0.6539	0.4641	0.6700	0.4840
RM3	0.7436	0.6467	0.4951	0.6945	0.4785
LDA	0.7169	0.6598	0.4771	0.6845	0.4993
PTM	0.7418	0.6618	0.5031	0.6944	0.5036
QUSTM	<b>0.7782</b> <sup>1</sup>	<b>0.7012</b> <sup>1,2,3,4</sup>	<b>0.5532</b> <sup>1,2,3,4</sup>	<b>0.7441</b> <sup>1,2,3,4</sup>	<b>0.5322</b> <sup>1,2,3,4</sup>
<i>ch%</i>	+5.68%	+6.92%	+20.97%	+10.88%	+11.88%

The highest value in each test set is marked in bold; superscripts 1,2,3 and 4 indicate statistically significant improvement at ( $p < 0.05$ ) over BM25, QL, KDE and LDA; and the *ch%* column denotes the improvements over QL.

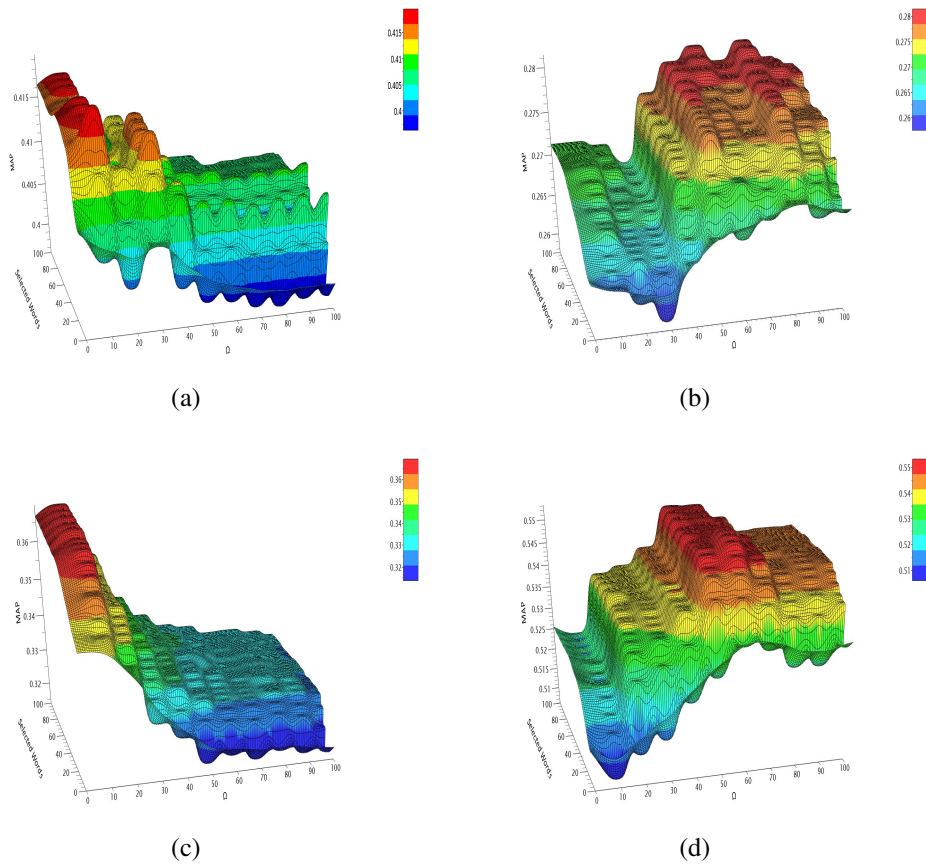
all microblog TREC dataset (2011-2014). Table 6.7 indicates that, for the MB11 test set, the proposed model improved over the P.30 by a maximum improvement of

13.98% compared to BM25 and enhanced by a 2.91% minimum compared to RM3. The proposed model improved over the MAP by a maximum of 24.68% compared to BM25 and a minimum of 11.18% compared to RM3. For the MB12 test set, the proposed model improved the P.30 by a maximum of 18.53% compared to BM25 and a 6.87% minimum compared to RM3. The proposed model developed the MAP by a maximum of 32.86% against compared to BM25 and an 11.05% minimum against compared to RM3.

To confirm the superiority of the proposed model, we tested the model on the Tweets2013 dataset, which is much larger than the Tweets2011 dataset, and showed the variations in performance. For the MB13 test set, Table 6.8 indicates that the proposed model improved the MAP by a maximum and minimum of 41.87% and 21.68% over BM25 and RM3, respectively. In addition, the corresponding increments of P.30 were a maximum of 23.84% and a minimum of 15.64% over BM25 and Recency, respectively. For the MB14 test set, the proposed model improved the P.30 by a maximum of 10.51% compared to BM25 and a minimum 5.95% compared to PTM. The proposed model developed the MAP by a maximum of 28.65% against compared to BM25 and a 9.96% minimum against compared to PTM.

To sum up the overall results, QUSTM shows significant improvement across test sets in both TREC microblog collections compared with the baseline models based on all reported metrics. The average improvement in MB11 is 10.32%, and in MB12 it reaches 14.92% improvement. For the second dataset, QUSTM has gained an average increase of 22.66% in MB13 and 10.42% in MB14. This result demonstrates that QUSTM is a valid hypothesis for social media search tasks.





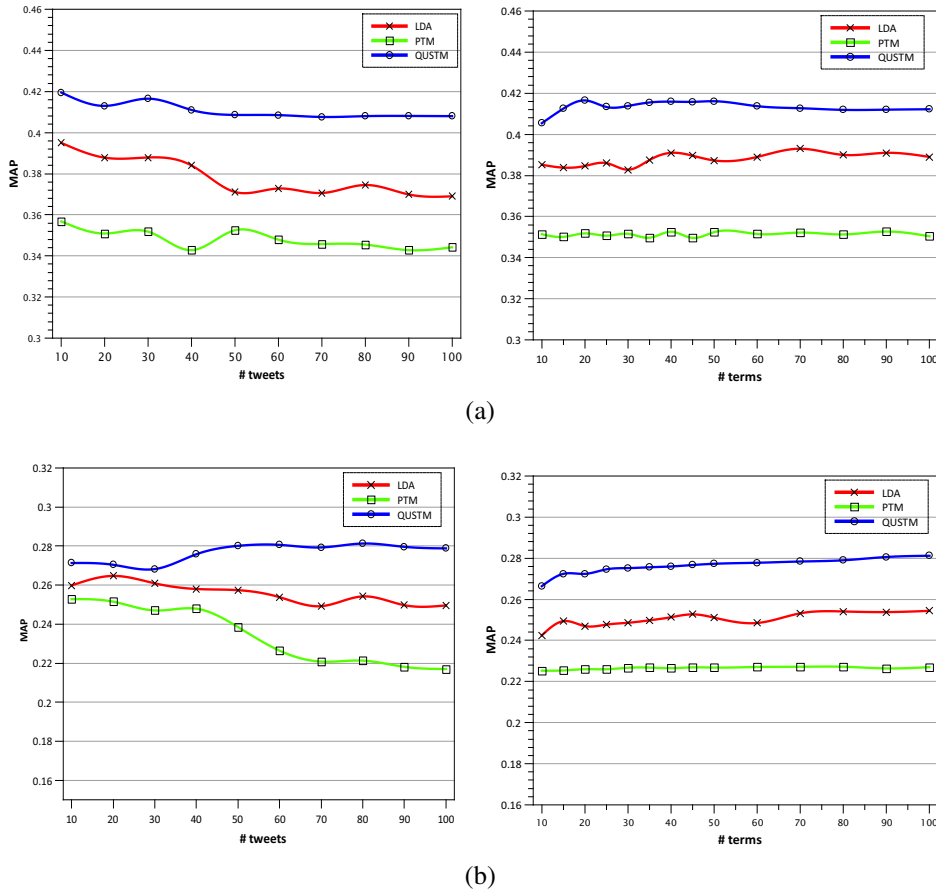
**Figure 6.12:** The proposed model parameters sensitivity

## 6.7.2 Discussion

In this section, we discuss the model results proposed for sensitivity parameters, compared with LDA, sensitivity to feedback and terms number and per-query analysis across all test sets.

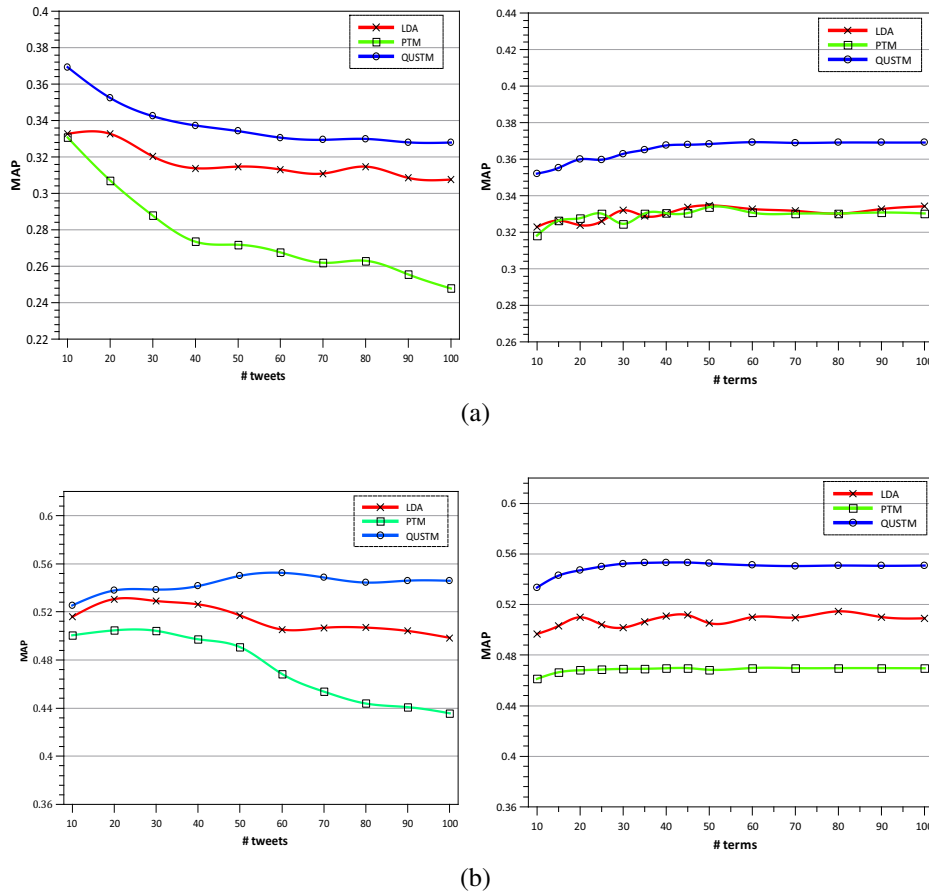
### 6.7.2.1 The proposed model sensitivity.

We show the proposed model's sensitivity to the number of selected tweets  $k$  that represent the input of the proposed model and the number of selected terms from  $\Omega$  in



**Figure 6.13:** The proposed model performance in terms from a selected number of tweets  $T$  and terms  $Q'$  for all test sets. (a) MB11 and (b) MB12

Figure 6.12, 6.13, and 6.14. An important issue that could face the social media search system's performance is the availability of relevant information in the selected tweets  $T$  and the terms. In order to mitigate this issue, we investigated the proposed model compared to the most robust baseline models that include (LDA and PTM). Figure 6.13 and 6.14 show the MAP performance with a different  $k$  value and terms across all test sets over the Tweets2011 and Tweets2013 collections. The  $k$  value of tweets set  $T$  is tested from 10 to 100 with an interval value of 10, and the number of terms that are used as a new representation of  $Q'$  is set from 10 to 100 with an interval value of 5.



**Figure 6.14:** The proposed model performance in terms from a selected number of tweets  $T$  and terms  $Q'$  for all test sets. (a) MB13 and (b) MB14.

It is clearly shown that the proposed model's performance is not sensitive to the change in the value of  $k$  or the terms number across the majority of test sets. On the other hand, the baseline model's performance dramatically decreases against the change of the value of  $k$ . Therefore, it proves the main assumption in this paper by reducing the uncertainty of information in the input of the search model.

**Table 6.9:** Results comparison.

Model	MB2011		MB2012		MB2013		MB2014	
	P30	MAP	P30	MAP	P30	MAP	P30	MAP
LDA	0.3902	0.3347	0.3586	0.2353	0.4504	0.2755	0.6598	0.4771
VRLDA	0.4020	0.4037	0.3845	0.2787	0.4937	0.3247	0.6877	0.5401
QUSTM	<b>0.4102</b>	<b>0.4127</b>	<b>0.3876</b>	<b>0.2814</b>	<b>0.5428</b>	<b>0.3693</b>	<b>0.7012</b>	<b>0.5532</b>

### 6.7.2.2 Compared with LDA

The proposed model has two major tasks: virtual document construction and term estimation. To verify the proposed query-based virtual document schema, we apply a virtual document to LDA, a state-of-the-art baseline model. Table 6.9 shows the proposed model QUSM compared with LDA and VRLDA (virtual document + LDA) over both datasets in all test sets. First, in LDA, the main input is the original retrieved tweets where tweets are individuals. Then, to prove the effectiveness of the proposed virtual document schema, we treat the input for LDA with our proposed virtual document schema. As Table 6.9 shows, VRLDA performance improved over P30 on average of 6.02% and significantly improved over MAP by 17.53% compared to LDA. This significant improvement on the LDA model, when the main input is virtual documents, indicates that there are high latent relationships between terms as we describe in Section 5.3.1. However, VRLDA still struggles to detect informative features that can describe the user’s information needs. The proposed model can reflect the discriminative for each candidate term in the virtual documents by estimating the accurate weight. As shown in Table 6.9, QUSTM significantly improved overall metrics and for all test sets in both datasets.

### 6.7.2.3 Per-query analysis.

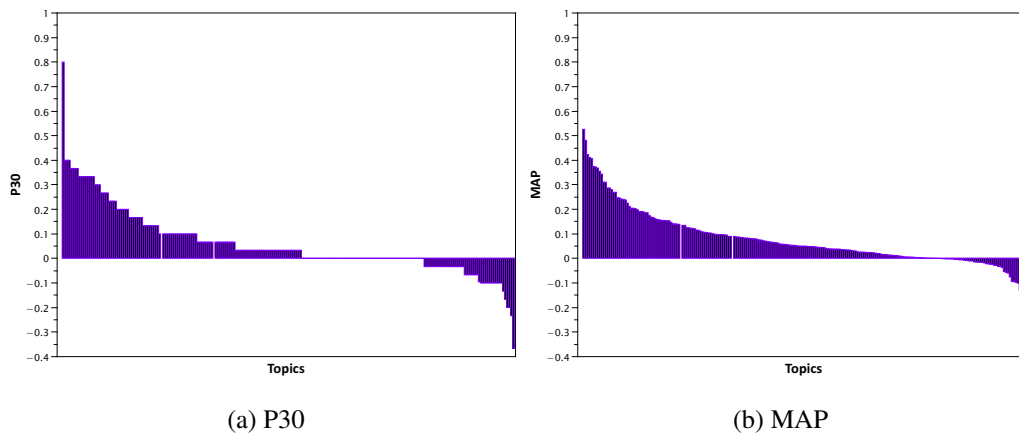
We conducted a comprehensive analysis of the improvements to the proposed model's performance over the baseline on a per-query base. Figure 6.15 shows the per-query improvement histogram for the proposed model performance compared with the QL baseline model over 224 queries for all test sets over *Tweets2011* and *Tweets2013* collections.

In practice, in MB2011, the proposed model performance wins on 34 queries and loses on 13 queries out of 49 queries; in MB2012, it wins on 51 queries and loses on 8 queries out of 60 queries. In MB2013, the proposed model performance wins on 48 queries and loses on 12 queries out of 60 queries; in MB2014, it wins on 43 queries and loses on 12 queries out of 55 queries. The average margin of the proposed model improvement regarding MAP evaluation metrics is also greater than the losses, at 78%.

In practice, in MB2011, the proposed model performance wins on 23 queries and loses on 10 queries out of 49 queries; in MB2012, it wins on 34 queries and loses on 11 queries out of 60 queries. In MB2013, the proposed model performance wins on 34 queries and loses on 9 queries out of 60 queries; in MB2014, it wins on 26 queries and loses on 15 queries out of 55 queries. The average margin of the proposed model improvement regarding P30 evaluation metrics is also greater than the losses, at 53%.

## 6.8 Compared Results With TREC

To further validate the effectiveness of the proposed methods, we compare (TAPRF, TBS and QUSTM) with the top five automatic runs in the TREC 2011, 2012, 2013 and 2014 microblog tracks Ounis et al. [2011], Soboroff et al. [2013, 2014, 2012]. Table 6.10 and 6.11 show the P@30 and MAP performances of all five runs and TREC's



**Figure 6.15:** Difference in the P30 and MAP between the proposed model and QL across all test sets.

baseline. TREC organisers ranked the submitted systems’ performances for TREC 2011 and 2012 based on the performance of P@30 (i.e. the official metric), while for TREC 2013 and 2014, MAP was the official metric.

As shown in Table 6.10 and 6.11, the proposed models are comparable with the top five runs in the TREC microblog track for topics (MB2011-MB2014). More specifically, for TREC 2011 topics (MB2011), the proposed models outperformed P@30 over the best-submitted result. TBS was ranked as 5<sup>th</sup> over 184 runs and improves MAP over the best submitted result by 55.29%. Meanwhile, for TREC 2012 topics (MB2012), the proposed models outperformed P@30 and MAP over the best submitted result. QUSTM was ranked 1<sup>st</sup> over 121 runs and improves P@30 and MAP over the best submitted result by 27.57% and 1.42%, respectively. Moreover, for TREC 2013 topics (MB2013), QUSTM ranked 1<sup>st</sup> and TBS ranked 2<sup>nd</sup> over 71 runs based P@30 metric. While QUSTM ranked 4<sup>st</sup> and TBS ranked 5<sup>th</sup> over 75 runs for TREC 2014 topics (MB2014). The main difference between the proposed models and the top five automatic runs is that TBS does not require any information from a linked document or external knowledge.

Compared with the TREC Microblog tasks (see competition results in [Ounis et al., 2011, Soboroff et al., 2013]), the proposed model QUSTM outperforms the majority of the best results. Specifically, for the MB2011 test set, QUSTM improves MAP over the best submitted system of the task by 54.09% and shows a 5.68% improvement for the MB2012 test set. QUSTM improves the MAP for the MB2013 test set over the best submitted system by 4.52%, while for the MB2014 test set the proposed model decreases the MAP over the best system by 6.41%. In order to determine the best system for the MB2014 test set, Lv Chao Fan and Yang [2014] employed the MB2013 test set as training set via RankSVM. They then utilised Google search engine's API to interpolate with local features, whereas the proposed models do not use any external data. Finally, for the MB2014 test set, QUSTM improves the MAP over the TREC baseline by 43.92%.

**Table 6.10:** The performance comparison of the proposed models TAPRF, TBS and QUSTM with the submitted TREC’s (2011-2012) microblog track runs.

<b>MB2011</b>				
Run	P30	MAP	docs?	external?
isiFDL	<b>0.4551</b>	0.1892	no	no
DFReeKLIM30	0.4401	0.2316	no	no
ri	0.4265	0.2203	no	yes
clarity1	0.4211	0.2109	no	no
FASILKOM02	0.4197	0.2082	no	yes
<i>baseline</i>	0.2925	0.1239	no	no
TAPRF	0.4020	0.3957	no	no
TBS	0.4204	<b>0.4249</b>	no	no
QUSTM	0.4102	0.4127	no	no
<b>MB2012</b>				
hitURLrun3	0.2701	0.2642	yes	no
hitLRrun1	0.2446	0.2411	no	no
ICTWDSERUN1	0.2384	0.2093	no	no
kobeL2R	0.2384	0.2081	no	no
hitDELMrun2	0.2350	0.2257	no	no
<i>baseline</i>	0.1390	0.1224	no	no
TAPRF	0.3638	0.2651	no	no
TBS	0.3757	0.2691	no	no
QUSTM	<b>0.3876</b>	<b>0.2814</b>	no	no

The best five automatic runs were presented beside the TREC *baseline* run. The column “docs?” indicates using linked documents and the column “external?” indicates using external information.



**Table 6.11:** The performance comparison of the proposed models TAPRF, TBS and QUSTM with the submitted TREC’s (2013-2014) microblog track runs.

MB2013				
Run	P30	MAP	docs?	external?
PrisRun4	0.5528	0.3524	yes	yes
QCRI4	0.5372	0.3494	yes	no
PKUICST3	<b>0.5567</b>	0.3486	yes	yes
PrisRun2	0.5511	0.3459	yes	no
PKUICST1	0.5478	0.3351	yes	yes
<i>baseline</i>	0.4500	0.2524	no	no
TAPRF	0.4839	0.3193	no	no
TBS	0.5228	0.3513	no	no
QUSTM	0.5428	<b>0.3693</b>	no	no
MB2014				
PKUICST3	<b>0.7224</b>	<b>0.5863</b>	yes	yes
hltcoe3	0.7121	0.5707	yes	yes
ECNURankLib	0.7133	0.5529	yes	no
PolyURun1	0.6994	0.5402	no	yes
ICARUN2	0.6909	0.5327	yes	yes
<i>baseline</i>	0.5145	0.3090	no	no
TAPRF	0.6836	0.5302	no	no
TBS	0.6903	0.5189	no	no
QUSTM	0.7012	0.5532	no	no

The best five automatic runs were presented beside the TREC *baseline* run. The column “docs?” indicates using linked documents and the column “external?” indicates using external information.

## 6.9 Summary

This chapter analyses the results of extensive experiments to evaluate the proposed TAPRF, TBS, and QUSTM methods. All of these methods attempt to discover the best

relevant features for expanding and reformulating user needs. First, an integration in the TAPRF between topical and lexical evidence was proposed. The proposed TBS method then identified a dynamic training set selection for a given query based on the topical distribution among the originally retrieved documents. Finally, the QUSTM model was used to model latent relationships to extract high quality features.

This chapter also describes the data collections, TREC microblog from 2011 to 2014 corpus (called Tweets2011 and Tweets2013) with four test sets, as our data sets are selected for evaluation as they come with a large number of short documents and relevant judgments. There are some differences between these two datasets, such as the number of tweets in each one. The data set for Tweets2013 includes more tweets than the data set for Tweets2011.

The results from this chapter compare the different state-of-the-art models using four standard metrics to evaluate the performance of the system. The TAPRF, TBS and QUSTM model results have been compared to lexical, topic and temporal based models. This research offers a promising method to evaluate high-quality features discovered in social media from short text documents. The results support the purposes of this study.

## Chapter 7

### Conclusions

---

Social media platforms, including Twitter and Facebook, have grown to be part of our daily lives. One of the most popular microblogging sites, Twitter, is estimated to accommodate around 320 million active users that produce approximately 500 million tweets per day. Due to the large volume and the velocity of published data, it is sometimes difficult for a user to find relevant information. This process can be tedious, and users might think that their search results are not available. The main objective of this research was to explore the problems affecting the discovery of relevant information on short text documents in social media data and argue the user information needs with relevant information.

Extracting informative features from short documents (e.g. microblogs) to meet a user's information needs is a challenging task in text mining and information retrieval. This problem allows researchers to explore alternative methods of extracting features based on individual user requirements (i.e., modification of classic retrieval models or expansion using an external knowledge base). To date, due to the nature of the microblog, many of these models still suffer from low quality, redundancy, and noise. We have also been studying and proposing several techniques to enhance a microblog

search. The thesis consisted of three significant contributions.

The main research issue addressed in this thesis is how to discover the relevant features from a set of pseudo-ranked short document features to improve social media search. This study evolves an effective topic-aware pseudo-relevance feedback model using a combination of well-established methodologies to find a set of relevant features closely related to the original user query. The discovery of latent topics is implemented as a method of representing hidden associations across terms in the given short documents using the LDA (Chapter 3). To evaluate the effectiveness of the proposed method, we conducted extensive experiments on the TREC microblog collections from 2011 to 2014. The obtained results show that our model outperforms the lexical-based baseline methods.

The second model of this thesis is based on the best training set to overcome the uncertainty of pseudo-document selection that is usually fixed for all queries. The proposed method views parameter  $k$  as a random variable (top- $k$  documents) and attempts to establish the best  $k$  value for each feedback document set that is pseudo-relevant to a query (Chapter 4). The proposed method consists of two stages. In the first stage, we automatically determine the top- $k$  documents for a particular query among the top documents. In particular, the random variable  $k$  arranges the top documents into different sub-sequences. We use LDA to determine how accurately the top  $k$  documents are used in selecting the best  $k$  value. In the second stage, we use the top- $k$  documents from the first stage to expand queries that use both latent and lexical features to select the relevant terms for the initial query. It also examines the temporal distribution of recent publications and provides a model for the efficient combination of lexical characteristics and latent issues. Our experiments on the TREC microblog datasets highlight the importance of dynamic pseudo feedback selection in ad-hoc microblog tasks. Results showed that further improvements are achieved through the

proposed model compared with fixed based pseudo feedback selection, such as RM3, and temporal-based models, such as KDE and Recency.

The ultimate goal of the third approach (Chapter 5) is to capture optimal implicit relationships from the initial tweets to infer more knowledge to serve user needs. The critical issue is how to reduce uncertainties in retrieved tweets in the absence of real user feedback. This approach proposed a new query-based, unsupervised discovery of relevant features to overcome LDA limitations when used with short text documents. We will receive the initial results immediately for a given query and select the top tweets. We then create a new virtual document space based on the tweets pooling strategy to query top-ranked tweets to discover a new link between user information needs and selected tweets. Then, concerning the relevant evidence, we retrieve a new weight for each word in the selected tweets. Experimental results based on the TREC microblog datasets demonstrated that the proposed model outperforms the baseline methods and shows promising results in comparison to topic model (e.g., LDA) and short-text-based topic models (e.g., PTM).

## 7.1 Synthesis of contributions

The following are the main contributions of this research study.

- **Effective topic-based PRF:** PRF is an alternative way to expand the user query without effort from the user. The primary input of the PRF model is the first-pass retrieved document set, where its quality varies. Extraction features from this set based on the classic retrieval model may introduce more noise terms to the original query. The new method is proposed to identify the relevant features

based on their topical distribution using LDA. The method uses linear interpolation to integrate the discovered topical evidence with the lexical evidence using a relevant model.

- **A new methodology to select training set:** It automatically estimates the representative pseudo-feedbacks used for a given query in the relevant feedback model, using topical distribution for the initial ranked documents. Furthermore, it integrated the topical distribution information from selected documents into the relevant feedback model to infer the discriminative power of each feature.
- **A novel query-based unsupervised learning method:** It is an unsupervised learning method that aims to capture implicit relationships that can increase short-text search performance in social media by addressing the sparsity problem. The basic idea behind the proposed model is to reduce the uncertainty of the tweets by augmentation using their content and then modeling the new association between terms via an intermediate set.

In short, three different models are presented in this research thesis:

1. The **TAPRF** method is a PRF-based topic capable of extracting relevant terms regarding their lexical and topical distribution.
2. The **TBS** is the topic-based training set selection method that automatically selected a set of dynamic pseudo-documents before applying the PRF model and then combined the selected topical features with the relevant model with respect to the temporal distribution between the tweet and query.
3. The **QUSTM** method is a novel technique for evoking self-augmentation for short text documents (called virtual documents) through query augmentation and

then modeling relationships to infer the power of discriminatives for the terms extracted.

## 7.2 Limitation and Future Directions

In this section, some potential future directions deriving from this research will be discussed. We focus on the main areas that we believe to be of the greatest significance.

- **In light of certain aspects of diversity in the search results:** To produce a search result for a given user query that includes various dimensions, in this research, we focused on the textual content of a stream tweets. However, it is worth examining other vital factors, as follows:

- *User content diversity:* Tweets from diverse sources, including official accounts of organisations' traditional media, celebrity accounts, and accounts of individual users (ordinary people) are more likely to meet the user information needs. This variation of sources reflects different viewpoints of the search results that can be accomplished using two different methodologies. First, personalise the search result based on the user profile that reflects their long-term intents. Second, user accounts can be categorised into different classes. This strategy can balance the number of tweets for each class, and the tweets that belong to the same class as the user can emphasise the user's needs. In this context, it is possible to examine both supervised and unsupervised learning techniques.

Second, user accounts can be categorised into different classes. This strategy can make a balance of the number of tweets for each class, and the tweets that belong to the same of the user class can be emphasis the user

needs. In this context, it is possible to examine both supervised and unsupervised learning techniques.

- *The diversity of viewpoint*: The user query could be about a debatable issue (e.g., political or tech topics), so it is interesting to include search results with different viewpoints. In such issues, the relevant information can appear on both sides. To address this problem, sentiment analysis and viewpoint discovery models can boost search results with more relevant information. Thus, different viewpoints or sentiments ensure the needs of the user are inclusive of various emotions or facts.
- *Geographical Evidence Diversity*: Twitter provides tweets with location information. The use of the geographical evidence factor can boost the variety of relevant information in returned search results. For example, when the user needs relate to a natural disaster, a tweet posted within the disaster-affected area is likely more relevant than a tweet posted out of the area of interest. The principal obstacle of introducing geographic evidence into the ad-hoc microblog task is to the diversity posts centralisation. Two different sources can indeed classify the tweet's location information. First, if the user has enabled the location information of their published tweets, we can get a precise geo-tagging feature information embed. Second, it can be approximated through the user's profile location.
- **Temporal distribution**: The first proposed model (**TAPRF**) assumes the temporal user information needs were uniformly distributed while in some queries are not. As future work, the model can extend through application to different time windows (e.g., a day, week, or month). For each time slot, individual search results can be produced for a user query with a specific time. A top  $k$  from each list can integrate into the initial list considering the time factor where



tweets in the recently ranked list to the user query can reword with appropriate weight. Thus, the main input for the proposed model will be the tweets from the integrated list.

- **Different schemes-based categorisation:** As to furthering direction of the second proposed model (**TBS**), we have only studied the dynamic selection of pseudo-documents to improve PRF performance with microblog data. Considering how to adopt expansion terms with the proposed model is an open research question for increasing PRF performance. Furthermore, the unique features of microblogs (such as hashtags or retweets) can be extended to further schemes that can be integrated with the proposed model and might improve microblog retrieval effectiveness. Another avenue for future work is the exploration of temporal distribution, in combination with the proposed model, to infer the relevance of features in the first pass retrieved documents that could be performed to enhance the user information needs.
- As to further directions of the third proposed model (**QUSTM**), we plan to integrate the temporal information with the evidence space in the proposed model. In addition, applying query performance predictors before applying the proposed model and interpolating with the number of user representation features provides an interesting direction in which to dynamically set the first-pass retrieved documents for each user's information need.
- In recent years, neural networks models have been offered a new effective paradigm to map and discover hidden relations between items (such as text features). Due to a semantic gap between queries and documents in a social media data, neural networks method, such as long short-term memory (LSTM) [Graves et al., 2008], word2vec [Mikolov et al., 2013], XLNET [Yang et al., 2019] and BERT

[Devlin et al., 2018] could offer a new light window for incorporating the proposed models for further investigations. Recently, pre-training neural networks models have been achieved great success with natural language processing applications in order to this kind of approaches have trained on massive amounts of label data.

## Appendix A

### TAPRF details results in the TREC microblog dataset

---

**Table A.1:** Details TAPRF Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
1	0.8000	0.6667	0.5770	0.8631	0.5556
2	0.6000	0.2000	0.2542	0.6081	0.2500
3	0.7000	0.7667	0.6737	0.8277	0.7667
4	0.6000	0.6000	0.3251	0.6601	0.4624
5	0.7000	0.3333	0.8258	0.9495	0.6364
6	0.2000	0.2000	0.1982	0.4818	0.1111
7	0.8000	0.8667	0.5143	0.7384	0.5463
8	0.8000	0.4667	0.4323	0.7487	0.5114
9	0.9000	0.9333	0.6306	0.8974	0.5969
10	0.2000	0.2667	0.2088	0.5391	0.2195
11	0.1000	0.1333	0.2067	0.5059	0.2000
12	0.1000	0.0667	0.2970	0.5552	0.2500
13	0.4000	0.3333	0.4418	0.6394	0.4348
14	1.0000	0.9333	0.5874	0.6793	0.5392
15	0.0000	0.0000	0.0000	0.0000	0.0000
16	0.1000	0.0333	0.5000	0.8262	0.5000

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
17	0.6000	0.4333	0.4297	0.7637	0.3696
18	0.1000	0.0333	1.0000	1.0000	1.0000
19	0.9000	0.7000	0.5337	0.7360	0.6200
20	0.8000	0.8333	0.6490	0.7357	0.6084
21	0.7000	0.6000	0.5643	0.7045	0.6071
22	0.6000	0.6667	0.4797	0.6802	0.5574
23	0.6000	0.5000	0.2629	0.4955	0.3855
24	0.4000	0.5000	0.2388	0.4626	0.4048
25	0.2000	0.4333	0.3446	0.5595	0.4800
26	0.9000	0.7667	0.3925	0.5598	0.4074
27	0.2000	0.1667	0.0479	0.1849	0.1277
28	0.4000	0.1333	0.4622	0.8044	0.5714
29	0.4000	0.5000	0.2313	0.5115	0.3021
30	0.9000	0.8333	0.5456	0.7611	0.6164
31	0.5000	0.2667	0.5487	0.8006	0.5000
32	0.2000	0.1333	0.0697	0.3911	0.1207
33	0.0000	0.0000	0.0000	0.0000	0.0000
34	0.4000	0.3667	0.2488	0.5468	0.3824
35	0.7000	0.3333	0.6824	0.8514	0.7000
36	0.7000	0.6667	0.5739	0.7292	0.6103
37	0.6000	0.8000	0.5821	0.8096	0.5405
37	0.6000	0.8000	0.5821	0.8096	0.5405
38	0.6000	0.3667	0.3939	0.7452	0.4500
39	0.4000	0.3667	0.3808	0.6977	0.3714
40	0.7000	0.3333	0.5404	0.9090	0.4706
41	0.4000	0.3667	0.3077	0.5215	0.3429
42	0.1000	0.0333	0.0654	0.4054	0.0385
43	0.6000	0.6000	0.5405	0.7852	0.5862
44	0.3000	0.2667	0.2382	0.4242	0.2727
45	0.1000	0.1333	0.0352	0.2598	0.1098
46	0.6000	0.2000	0.5704	0.7276	0.6667
47	0.0000	0.0667	0.0170	0.1210	0.0000
48	0.4000	0.4667	0.2348	0.4779	0.4444
49	0.1000	0.0333	0.5031	0.6868	0.5000
51	0.0000	0.0000	0.0007	0.0312	0.0000
52	0.7000	0.5000	0.3786	0.4774	0.4839
53	0.0000	0.0000	0.0031	0.1197	0.0000

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
54	1.0000	0.8333	0.5843	0.8135	0.5850
55	1.0000	1.0000	0.8567	0.9328	0.8235
56	0.8000	0.4667	0.4672	0.8765	0.4643
57	0.7000	0.5333	0.2275	0.4605	0.3218
58	0.0000	0.0000	0.1090	0.4370	0.0000
59	0.0000	0.0000	0.0468	0.3544	0.0000
60	0.0000	0.0000	0.1737	0.4563	0.3362
61	0.7000	0.2667	0.2487	0.4328	0.2759
62	0.5000	0.6000	0.5531	0.8605	0.6000
63	0.3000	0.2000	0.1349	0.3559	0.3333
64	0.9000	0.7000	0.5191	0.7041	0.6078
65	0.3000	0.3000	0.2031	0.5179	0.2812
66	0.5000	0.4000	0.2928	0.6890	0.3462
67	0.0000	0.0667	0.0787	0.3844	0.0417
68	0.9000	0.6667	0.4423	0.6934	0.4357
69	0.1000	0.0667	0.0221	0.2444	0.0500
70	0.4000	0.3333	0.3042	0.6068	0.3077
71	0.8000	0.6000	0.5059	0.7970	0.5461
72	0.0000	0.1667	0.0952	0.4064	0.1268
73	0.9000	0.7667	0.4283	0.7927	0.4844
74	0.2000	0.2333	0.2054	0.5316	0.3467
75	0.5000	0.4000	0.4157	0.7556	0.5029
77	0.1000	0.1333	0.0533	0.2204	0.1538
78	0.5000	0.6333	0.2101	0.5114	0.3201
79	0.7000	0.5000	0.3760	0.7111	0.4817
80	0.1000	0.0333	0.0681	0.3570	0.0526
81	0.7000	0.4000	0.2673	0.4897	0.3505
82	0.3000	0.3333	0.2439	0.6084	0.3936
83	0.4000	0.5000	0.2831	0.5210	0.5185
84	0.7000	0.4000	0.3715	0.7061	0.4286
85	0.0000	0.0333	0.0177	0.2891	0.0167
86	0.8000	0.5333	0.6124	0.8498	0.6400
87	0.9000	0.8667	0.3648	0.5438	0.4133
88	1.0000	0.9667	0.5492	0.7844	0.5615
89	0.0000	0.0000	0.0501	0.3293	0.0000
90	0.5000	0.3333	0.1938	0.6044	0.2333
91	0.3000	0.1333	0.1283	0.4190	0.1304

---

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
92	0.5000	0.4000	0.2876	0.5490	0.2955
93	0.1000	0.1000	0.0735	0.3771	0.1081
94	0.5000	0.3000	0.1864	0.4607	0.2500
95	1.0000	0.8000	0.4621	0.7064	0.5435
96	0.1000	0.0333	0.1244	0.5399	0.0789
97	0.5000	0.4667	0.1348	0.3241	0.2444
98	0.9000	0.6333	0.3010	0.5699	0.3284
99	0.2000	0.2667	0.0530	0.2577	0.1723
100	0.3000	0.1333	0.2446	0.4868	0.2308
101	0.4000	0.3000	0.2682	0.6018	0.2941
102	0.2000	0.1000	0.0554	0.2070	0.1250
103	0.6000	0.6667	0.5808	0.6060	0.7308
104	0.4000	0.4667	0.3237	0.6127	0.4286
105	0.0000	0.0000	0.0221	0.2689	0.0000
106	0.4000	0.2667	0.2757	0.6402	0.2885
107	0.0000	0.0000	0.0258	0.1332	0.1392
108	0.9000	0.7333	0.4280	0.6835	0.4933
109	0.5000	0.3333	0.2439	0.5583	0.3913
110	0.8000	0.5667	0.4662	0.6677	0.4722
111	0.2000	0.1667	0.0388	0.1475	0.1528
112	0.2000	0.0667	0.1358	0.5850	0.0952
113	0.8000	0.2667	0.2272	0.4116	0.2500
114	0.3000	0.1000	0.0680	0.2618	0.1304
115	0.5000	0.4000	0.2485	0.6008	0.2623
116	0.9000	0.8333	0.4297	0.7233	0.4767
117	0.1000	0.1333	0.0898	0.3670	0.1333
118	0.0000	0.4000	0.2969	0.4369	0.4534
119	1.0000	0.5333	0.4054	0.6516	0.4324
120	1.0000	0.4000	0.2338	0.3046	0.2353
121	0.7000	0.7333	0.6117	0.7546	0.6491
122	0.0000	0.0000	0.0044	0.1137	0.0000
123	1.0000	0.9000	0.2546	0.4602	0.3099
124	0.1000	0.0333	0.0637	0.3909	0.0312
125	0.8000	0.8333	0.2926	0.6862	0.2774
126	1.0000	0.9333	0.3532	0.5915	0.4143
127	1.0000	0.9667	0.7379	0.7997	0.7014
128	1.0000	0.9667	0.3679	0.6624	0.3843

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
129	1.0000	1.0000	0.7839	0.9266	0.7500
130	0.4000	0.2667	0.1563	0.4737	0.3137
131	0.6000	0.3333	0.3522	0.5615	0.4375
132	0.4000	0.1667	0.0799	0.2870	0.2236
133	1.0000	0.7000	0.4952	0.6936	0.5327
134	0.0000	0.0667	0.0059	0.0830	0.0690
135	0.4000	0.6667	0.3209	0.6406	0.4020
136	0.3000	0.1000	0.0238	0.1089	0.0286
137	0.2000	0.2667	0.0801	0.3258	0.1745
138	1.0000	1.0000	0.7370	0.8946	0.7039
139	1.0000	0.6333	0.4801	0.7774	0.5000
140	0.9000	0.5667	0.2351	0.6309	0.2190
141	1.0000	1.0000	0.6121	0.6983	0.6447
142	0.5000	0.3333	0.1708	0.5678	0.1546
143	1.0000	0.5333	0.4766	0.6741	0.3810
144	0.7000	0.7667	0.2671	0.5558	0.2439
145	0.9000	0.3000	0.1645	0.5357	0.2674
146	1.0000	1.0000	0.6620	0.8367	0.6327
147	0.3000	0.4333	0.4427	0.6909	0.5286
148	0.4000	0.2000	0.1155	0.4284	0.1429
149	1.0000	0.6667	0.3332	0.5638	0.3625
150	0.0000	0.0000	0.0007	0.0168	0.0000
151	0.0000	0.0000	0.0199	0.1509	0.0000
152	1.0000	0.7333	0.7145	0.8401	0.7857
153	0.8000	0.3333	0.1960	0.6336	0.2317
154	0.5000	0.6333	0.4093	0.6252	0.5135
155	0.4000	0.2667	0.2454	0.5865	0.2857
156	1.0000	0.9667	0.7616	0.8065	0.7979
157	1.0000	1.0000	0.4444	0.6096	0.4430
158	0.0000	0.0000	0.1738	0.4710	0.2133
159	0.8000	0.6333	0.7407	0.8643	0.8636
160	0.0000	0.0000	0.0211	0.2412	0.0000
161	0.4000	0.4667	0.2149	0.5715	0.3058
162	0.4000	0.4333	0.2569	0.6060	0.3074
163	1.0000	1.0000	0.6940	0.8117	0.6776
164	0.4000	0.3333	0.2626	0.6358	0.1872
165	0.1000	0.0333	0.1436	0.3067	0.1429

---

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
166	0.5000	0.3333	0.6574	0.8673	0.5385
167	0.1000	0.2333	0.0582	0.3421	0.1349
168	0.2000	0.2333	0.0982	0.3451	0.2353
169	1.0000	0.9667	0.8198	0.9429	0.8000
170	1.0000	0.7667	0.3673	0.6594	0.3594
171	0.7000	0.8667	0.5805	0.8747	0.4792
172	1.0000	1.0000	0.8630	0.9478	0.8201
173	0.1000	0.0333	0.0055	0.0739	0.0294
174	0.8000	0.2667	0.5739	0.8081	0.5333
175	1.0000	0.7667	0.5323	0.7416	0.5672
176	0.3000	0.1000	0.0159	0.1100	0.0312
177	0.9000	0.9667	0.4005	0.7888	0.3333
178	0.9000	0.9667	0.6297	0.8310	0.6343
179	0.0000	0.1000	0.2215	0.5734	0.1000
180	0.5000	0.8333	0.4556	0.6086	0.5325
181	1.0000	0.6333	0.6797	0.8542	0.6552
182	1.0000	0.8667	0.7598	0.9272	0.7117
183	1.0000	0.9667	0.9441	0.9803	0.8785
184	1.0000	0.8333	0.3764	0.7073	0.3669
185	1.0000	0.6667	0.8881	0.9537	0.8182
186	1.0000	0.6000	0.6038	0.8665	0.5000
187	1.0000	0.9000	0.8786	0.9673	0.8293
188	1.0000	0.5667	0.6451	0.9043	0.5143
189	0.0000	0.0000	0.0000	0.0176	0.0000
190	1.0000	1.0000	0.8364	0.9583	0.7761
191	1.0000	0.9667	0.7439	0.9099	0.6930
192	0.8000	0.8333	0.5611	0.6419	0.6061
193	0.0000	0.3333	0.3613	0.5731	0.4907
194	0.0000	0.1000	0.0888	0.4176	0.1250
195	1.0000	0.9000	0.4813	0.6872	0.5337
196	0.9000	0.9333	0.5404	0.7772	0.5537
197	1.0000	0.9333	0.6065	0.8490	0.5631
198	1.0000	0.8667	0.5190	0.6266	0.5373
199	1.0000	1.0000	0.8995	0.9584	0.8112
200	1.0000	0.8333	0.4352	0.7588	0.4896
201	1.0000	0.8333	0.7741	0.8876	0.7264
202	1.0000	1.0000	0.8886	0.9657	0.8643

Continued on next page



---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
203	0.4000	0.2333	0.0395	0.1461	0.0909
204	0.9000	0.7667	0.6959	0.8471	0.7419
205	1.0000	0.9667	0.6441	0.9201	0.6471
206	1.0000	0.9000	0.4245	0.8399	0.3766
207	0.9000	0.9000	0.6451	0.8393	0.6695
208	1.0000	1.0000	0.8771	0.9232	0.8039
209	1.0000	1.0000	0.5456	0.6940	0.5228
210	0.6000	0.4667	0.4859	0.8897	0.4688
211	0.6000	0.4000	0.4429	0.7971	0.5000
212	1.0000	0.6333	0.3914	0.8113	0.3431
213	1.0000	1.0000	0.9416	0.9798	0.9161
214	0.7000	0.5000	0.5144	0.7835	0.4967
215	1.0000	0.9667	0.6169	0.6388	0.6594
216	0.7000	0.3000	0.2186	0.6202	0.2412
217	0.9000	0.7667	0.1814	0.3416	0.2513
218	1.0000	0.9667	0.8421	0.9436	0.7645
219	0.3000	0.1000	0.1252	0.4846	0.1071
220	0.3000	0.1333	0.0915	0.3168	0.1429
221	1.0000	1.0000	0.8242	0.9401	0.7752
222	1.0000	0.9667	0.5403	0.5983	0.5525
223	0.5000	0.3000	0.1063	0.3630	0.1622
224	0.5000	0.3000	0.5175	0.8733	0.3846
225	0.8000	0.5667	0.6558	0.8056	0.6216

---



## Appendix B

### TBS details results in the TREC microblog dataset

---

**Table B.1:** Details TBS Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
1	0.9000	0.7667	0.6259	0.8614	0.5926
2	0.9000	0.5000	0.5924	0.8708	0.5833
3	0.7000	0.7333	0.7236	0.8853	0.7333
4	0.6000	0.6333	0.3129	0.6553	0.3548
5	0.9000	0.3333	0.8822	0.9620	0.8182
6	0.5000	0.2000	0.5818	0.8143	0.5556
7	0.9000	0.9000	0.5209	0.7133	0.5463
8	0.8000	0.5667	0.4291	0.7441	0.4545
9	0.9000	0.9333	0.6295	0.8986	0.6047
10	0.1000	0.2667	0.1704	0.5288	0.2927
11	0.1000	0.1333	0.2156	0.4997	0.2000
12	0.2000	0.1000	0.5547	0.7766	0.5000
13	0.5000	0.4000	0.3430	0.6009	0.4783
14	1.0000	0.9333	0.5597	0.6886	0.5294
15	0.0000	0.0000	0.0000	0.0000	0.0000
16	0.1000	0.0667	0.5625	0.8532	0.5000

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
17	0.4000	0.3667	0.3833	0.7227	0.3478
18	0.1000	0.0333	1.0000	1.0000	1.0000
19	0.8000	0.7667	0.5483	0.7495	0.6400
20	0.7000	0.7667	0.5869	0.7847	0.5734
21	0.6000	0.7000	0.5434	0.7721	0.5429
22	0.7000	0.7333	0.5718	0.7550	0.5738
23	0.4000	0.4667	0.2133	0.4681	0.3614
24	0.5000	0.6667	0.3170	0.6042	0.4167
25	0.2000	0.5000	0.2959	0.5827	0.4267
26	1.0000	0.6333	0.4019	0.6602	0.4259
27	0.2000	0.1333	0.0434	0.1787	0.1277
28	0.4000	0.1333	0.4622	0.6300	0.5714
29	0.4000	0.3667	0.1993	0.5742	0.2292
30	0.6000	0.5667	0.4611	0.7125	0.5479
31	0.5000	0.3000	0.6194	0.8634	0.5000
32	0.0000	0.0333	0.0288	0.2526	0.0517
33	0.0000	0.0000	0.0000	0.0000	0.0000
34	0.4000	0.4000	0.2575	0.5785	0.3824
35	0.7000	0.3333	0.6958	0.8657	0.7000
36	0.8000	0.7000	0.5916	0.8033	0.5956
37	0.7000	0.7667	0.5808	0.8331	0.5270
38	0.7000	0.3333	0.4476	0.7918	0.5000
39	0.3000	0.4000	0.3564	0.6722	0.3714
40	0.7000	0.3667	0.5034	0.8467	0.4706
41	0.3000	0.3667	0.3241	0.5841	0.4286
42	0.3000	0.3333	0.1851	0.4712	0.3846
43	0.6000	0.6000	0.5610	0.7986	0.6207
44	0.6000	0.5000	0.5587	0.6985	0.5909
45	0.4000	0.2333	0.0705	0.3271	0.1098
46	0.6000	0.2000	0.5849	0.7454	0.6667
47	0.1000	0.0667	0.0418	0.1953	0.1538
48	0.5000	0.3333	0.1758	0.4318	0.3519
49	0.1000	0.0333	0.5034	0.6880	0.5000
51	0.0000	0.0000	0.0006	0.0273	0.0000
52	0.7000	0.5000	0.3529	0.5453	0.4839
53	0.0000	0.0000	0.0013	0.1042	0.0000
54	0.7000	0.7333	0.5060	0.7373	0.5882

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
55	1.0000	1.0000	0.8466	0.9331	0.8162
56	0.8000	0.5000	0.4597	0.8367	0.5000
57	0.6000	0.4667	0.2110	0.4771	0.3333
58	0.0000	0.0000	0.0204	0.2137	0.0000
59	0.0000	0.0667	0.0505	0.4008	0.0638
60	0.0000	0.0333	0.1590	0.4300	0.3341
61	0.6000	0.3667	0.2790	0.5234	0.3448
62	0.6000	0.5667	0.5661	0.8574	0.5667
63	0.4000	0.2000	0.2698	0.5264	0.4167
64	0.9000	0.7000	0.5236	0.7265	0.6275
65	0.4000	0.3667	0.3119	0.6717	0.3750
66	0.6000	0.5000	0.2581	0.6091	0.3654
67	0.0000	0.0667	0.0921	0.4085	0.0833
68	0.7000	0.4667	0.4076	0.7298	0.4286
69	0.2000	0.1000	0.0301	0.2810	0.0500
70	0.3000	0.3000	0.3098	0.6608	0.3538
71	0.8000	0.7000	0.4466	0.8003	0.4894
72	0.1000	0.2333	0.1046	0.4406	0.1408
73	0.7000	0.7667	0.3865	0.7217	0.4844
74	0.4000	0.5000	0.3054	0.6716	0.3400
75	0.5000	0.4333	0.3930	0.7570	0.4393
77	0.0000	0.1667	0.0538	0.2244	0.1731
78	0.4000	0.5667	0.2533	0.5598	0.3598
79	0.6000	0.4667	0.3477	0.7005	0.4207
80	0.4000	0.2333	0.2009	0.5073	0.3684
81	0.0000	0.2667	0.1600	0.4975	0.1682
82	0.4000	0.4000	0.2574	0.6198	0.3830
83	0.4000	0.4333	0.3396	0.6496	0.4444
84	0.7000	0.5000	0.4172	0.7387	0.4286
85	0.0000	0.0000	0.0198	0.3005	0.0167
86	0.8000	0.7333	0.7490	0.8986	0.8000
87	0.9000	0.7667	0.3524	0.5407	0.4067
88	0.9000	0.9000	0.5636	0.7905	0.5779
89	0.0000	0.0333	0.0491	0.3387	0.0000
90	0.5000	0.3333	0.1930	0.6008	0.2333
91	0.3000	0.1667	0.1654	0.5048	0.1739
92	0.4000	0.3000	0.1760	0.4661	0.2727

---

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
93	0.0000	0.1000	0.0505	0.3607	0.0811
94	0.4000	0.3333	0.1867	0.5005	0.2273
95	1.0000	0.8000	0.4422	0.6980	0.5362
96	0.1000	0.0333	0.1714	0.5989	0.2105
97	0.5000	0.4667	0.1760	0.4808	0.2667
98	0.7000	0.5333	0.2311	0.5498	0.3209
99	0.4000	0.4667	0.1063	0.3569	0.1892
100	0.3000	0.1333	0.2512	0.4696	0.2308
101	0.3000	0.3333	0.2934	0.6932	0.2941
102	0.1000	0.1000	0.0580	0.2507	0.0833
103	0.7000	0.6667	0.6001	0.6611	0.7692
104	0.4000	0.5000	0.3171	0.6132	0.4107
105	0.0000	0.0333	0.0216	0.2704	0.0000
106	0.4000	0.5333	0.3743	0.7202	0.4615
107	0.1000	0.1333	0.1022	0.4418	0.1266
108	0.8000	0.6333	0.4154	0.6927	0.4533
109	0.5000	0.3000	0.2537	0.5729	0.3913
110	0.2000	0.2333	0.2352	0.5262	0.3056
111	0.3000	0.2333	0.0580	0.1990	0.1528
112	0.8000	0.3000	0.4345	0.7826	0.4286
113	0.8000	0.2667	0.2431	0.4215	0.2500
114	0.4000	0.1667	0.1418	0.4006	0.2174
115	0.4000	0.4000	0.2674	0.6421	0.3115
116	0.9000	0.8000	0.3460	0.6478	0.4109
117	0.6000	0.3000	0.2536	0.5639	0.3000
118	0.8000	0.8333	0.4373	0.6047	0.5127
119	1.0000	0.5333	0.5073	0.7910	0.4595
120	0.9000	0.6333	0.3727	0.6252	0.4118
121	0.7000	0.7667	0.6240	0.7863	0.6667
122	0.0000	0.0000	0.0033	0.1039	0.0000
123	0.9000	0.9000	0.2397	0.4157	0.3041
124	0.1000	0.0333	0.0260	0.2453	0.0590
125	0.9000	0.7000	0.2782	0.6682	0.3358
126	1.0000	0.9333	0.3377	0.5592	0.4011
127	1.0000	1.0000	0.5586	0.7241	0.5540
128	1.0000	0.9000	0.3734	0.6354	0.4202
129	1.0000	1.0000	0.8311	0.9442	0.7414

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
130	0.4000	0.2000	0.1565	0.5721	0.2353
131	0.8000	0.4333	0.6881	0.8980	0.6250
132	0.3000	0.1667	0.0850	0.3102	0.2236
133	1.0000	0.9000	0.3253	0.5611	0.4402
134	0.0000	0.0000	0.0040	0.0774	0.0431
135	0.9000	0.7667	0.4045	0.6948	0.5196
136	0.3000	0.2000	0.1052	0.3751	0.2381
137	0.4000	0.2667	0.1029	0.3859	0.1946
138	1.0000	1.0000	0.6834	0.8731	0.6776
139	1.0000	0.7667	0.4839	0.8071	0.4630
140	0.9000	0.6667	0.3070	0.6398	0.3212
141	1.0000	0.9667	0.6071	0.7054	0.6151
142	1.0000	0.4000	0.2931	0.6475	0.2990
143	1.0000	0.6667	0.5967	0.7657	0.5476
144	0.9000	0.6667	0.4050	0.7036	0.4573
145	0.7000	0.4667	0.2061	0.5730	0.2907
146	1.0000	1.0000	0.6458	0.7621	0.6182
147	0.3000	0.6000	0.4850	0.7183	0.6000
148	0.6000	0.3000	0.1962	0.5536	0.1837
149	0.9000	0.5000	0.3352	0.6749	0.3375
150	0.0000	0.0000	0.0007	0.0239	0.0000
151	0.0000	0.0000	0.0232	0.1699	0.0000
152	0.7000	0.4333	0.3841	0.6123	0.4643
153	0.9000	0.6000	0.3972	0.7840	0.4390
154	0.4000	0.4000	0.2966	0.5810	0.4324
155	0.3000	0.3333	0.1027	0.3563	0.2727
156	1.0000	1.0000	0.8900	0.9437	0.8830
157	1.0000	1.0000	0.4943	0.6844	0.4832
158	0.0000	0.0667	0.1724	0.4819	0.2000
159	0.8000	0.6333	0.7150	0.8407	0.8182
160	0.0000	0.0333	0.0356	0.3094	0.0455
161	0.4000	0.7000	0.3084	0.6089	0.3425
162	0.6000	0.6333	0.3725	0.6928	0.4459
163	1.0000	1.0000	0.7178	0.8442	0.6869
164	0.1000	0.0333	0.1936	0.6052	0.0321
165	0.2000	0.0667	0.2000	0.3812	0.2857
166	0.6000	0.3667	0.7363	0.9052	0.6923

---

Continued on next page

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
167	0.2000	0.2333	0.0911	0.4355	0.1587
168	0.3000	0.3667	0.2231	0.5650	0.2794
169	1.0000	1.0000	0.8395	0.9603	0.7455
170	1.0000	0.8333	0.4339	0.7739	0.4062
171	0.7000	0.8667	0.6162	0.8614	0.5938
172	1.0000	1.0000	0.8021	0.9165	0.7835
173	0.1000	0.2000	0.1442	0.4827	0.1765
174	0.8000	0.3667	0.7175	0.9235	0.5333
175	1.0000	1.0000	0.4249	0.7196	0.4030
176	0.3000	0.1000	0.0157	0.1499	0.0521
177	1.0000	0.9667	0.7756	0.9240	0.6736
178	1.0000	0.9667	0.6921	0.8660	0.6306
179	0.0000	0.1000	0.1605	0.5458	0.1250
180	0.8000	0.7667	0.5152	0.6798	0.5753
181	1.0000	0.6333	0.6416	0.7617	0.6552
182	0.9000	0.8333	0.6574	0.9094	0.6260
183	1.0000	0.9667	0.8877	0.9663	0.8542
184	1.0000	0.9000	0.4098	0.7472	0.3964
185	1.0000	0.6667	0.9118	0.9490	0.8636
186	0.5000	0.5333	0.3141	0.5555	0.4444
187	1.0000	0.8333	0.8048	0.8944	0.7805
188	1.0000	0.5667	0.6464	0.9048	0.5143
189	0.0000	0.0000	0.0008	0.0449	0.0000
190	1.0000	1.0000	0.7924	0.9399	0.7463
191	1.0000	0.9333	0.6642	0.8963	0.6228
192	0.8000	0.8333	0.5748	0.6983	0.6061
193	0.1000	0.1000	0.1534	0.5333	0.1204
194	0.0000	0.1667	0.1261	0.4515	0.1875
195	1.0000	0.8000	0.4850	0.7467	0.5112
196	1.0000	0.9333	0.8240	0.9481	0.7851
197	1.0000	1.0000	0.6273	0.8277	0.5922
198	0.3000	0.5667	0.2984	0.5762	0.4328
199	1.0000	1.0000	0.8836	0.9597	0.8089
200	0.8000	0.8333	0.5005	0.7968	0.5729
201	1.0000	1.0000	0.7438	0.8534	0.7752
202	0.9000	0.9667	0.8760	0.9625	0.8643
203	0.2000	0.2333	0.0288	0.1725	0.0909

Continued on next page



---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
204	0.9000	0.6667	0.5641	0.7636	0.6452
205	1.0000	1.0000	0.6359	0.8705	0.5928
206	1.0000	0.8667	0.4158	0.8205	0.3636
207	0.9000	0.9000	0.7020	0.8371	0.6356
208	1.0000	0.9333	0.7804	0.9435	0.7426
209	1.0000	1.0000	0.5207	0.6603	0.5277
210	0.5000	0.5000	0.3581	0.7099	0.4688
211	0.8000	0.6000	0.6278	0.8518	0.6250
212	1.0000	0.6667	0.4004	0.8252	0.3723
213	1.0000	1.0000	0.9132	0.9855	0.8504
214	0.8000	0.4333	0.4674	0.7970	0.4641
215	1.0000	1.0000	0.6154	0.6636	0.6594
216	0.1000	0.2333	0.1701	0.5225	0.2647
217	0.8000	0.8000	0.1737	0.3840	0.2154
218	1.0000	0.9667	0.8242	0.9382	0.7864
219	0.5000	0.1667	0.1935	0.4686	0.1786
220	0.3000	0.1667	0.1625	0.4342	0.1786
221	1.0000	1.0000	0.4868	0.7158	0.5302
222	1.0000	0.9667	0.3207	0.3601	0.2983
223	0.5000	0.3333	0.1067	0.3873	0.1892
224	0.7000	0.2667	0.5693	0.8663	0.5385
225	0.9000	0.8667	0.8147	0.8925	0.7838

---



## Appendix C

### QUSTM details results in the TREC microblog dataset

---

**Table C.1:** Details QUSTM Evaluation Result on the TREC microblog dataset over all test sets MB2011 to MB2014

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
1	0.8000	0.6667	0.5998	0.8158	0.5741
2	0.1000	0.0667	0.0975	0.4138	0.0833
3	0.7000	0.8000	0.7263	0.8369	0.8000
4	0.7000	0.6333	0.3718	0.6898	0.4516
5	0.7000	0.3333	0.8411	0.9540	0.7273
6	0.2000	0.2333	0.2081	0.4641	0.1111
7	0.8000	0.8667	0.5516	0.7412	0.5926
8	0.9000	0.7667	0.4926	0.7796	0.4545
9	0.9000	0.9333	0.6304	0.8990	0.6047
10	0.2000	0.2333	0.2511	0.5637	0.2195
11	0.2000	0.1333	0.2426	0.5506	0.2000
12	0.2000	0.1000	0.3375	0.5923	0.2500
13	0.5000	0.4000	0.4404	0.6356	0.4783
14	1.0000	0.8000	0.5029	0.6076	0.5000
15	0.0000	0.0000	0.0000	0.0000	0.0000
16	0.1000	0.0667	0.5833	0.9007	0.5000

Continued on next page

Q	P10	P30	MAP	NDCG	Rprec
17	0.5000	0.2667	0.3757	0.7021	0.3478
18	0.1000	0.0333	1.0000	1.0000	1.0000
19	0.9000	0.7333	0.5481	0.7148	0.6400
20	0.8000	0.8667	0.6438	0.7370	0.6014
21	0.7000	0.6000	0.5699	0.7304	0.5714
22	0.9000	0.7667	0.5657	0.7308	0.5820
23	0.4000	0.5000	0.2459	0.4898	0.3735
24	0.5000	0.7000	0.3093	0.5174	0.4524
25	0.1000	0.4000	0.3051	0.5423	0.4667
26	1.0000	0.6667	0.3978	0.6125	0.4167
27	0.3000	0.1667	0.0731	0.2570	0.1489
28	0.4000	0.1333	0.4930	0.5978	0.5714
29	0.4000	0.4333	0.2124	0.5060	0.2604
30	0.8000	0.7667	0.5352	0.6833	0.6027
31	0.5000	0.2667	0.6340	0.8664	0.5000
32	0.1000	0.1333	0.0568	0.3350	0.1379
33	0.0000	0.0000	0.0000	0.0000	0.0000
34	0.4000	0.4000	0.2349	0.5341	0.3529
35	0.7000	0.3333	0.6976	0.8599	0.7000
36	0.8000	0.6000	0.5916	0.7525	0.5882
37	0.7000	0.7667	0.5874	0.8156	0.5405
38	0.7000	0.4000	0.4786	0.8447	0.4500
39	0.2000	0.3667	0.3542	0.6899	0.4000
40	0.7000	0.4000	0.5866	0.9222	0.5294
41	0.4000	0.3667	0.3594	0.5581	0.3429
42	0.2000	0.1667	0.1181	0.3842	0.1923
43	0.5000	0.5667	0.5357	0.8083	0.5517
44	0.6000	0.5000	0.5436	0.6317	0.5909
45	0.3000	0.1333	0.0556	0.3055	0.1341
46	0.6000	0.2000	0.5571	0.7648	0.6667
47	0.0000	0.0667	0.0126	0.0880	0.0000
48	0.3000	0.3333	0.1607	0.4117	0.3519
49	0.1000	0.0333	0.5034	0.6880	0.5000
51	0.0000	0.0000	0.0014	0.0399	0.0377
52	0.9000	0.5000	0.4047	0.5122	0.4839
53	0.0000	0.0000	0.0013	0.1048	0.0000
54	1.0000	0.9000	0.6344	0.8149	0.6405

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
55	1.0000	1.0000	0.8824	0.9425	0.8456
56	0.8000	0.4667	0.4814	0.8797	0.4643
57	0.8000	0.6000	0.2555	0.4854	0.3333
58	0.0000	0.0000	0.0996	0.4383	0.0000
59	0.0000	0.0000	0.0432	0.3511	0.0000
60	0.0000	0.2333	0.2138	0.5173	0.3905
61	0.7000	0.4333	0.3295	0.4981	0.4483
62	0.6000	0.6000	0.5740	0.8750	0.6000
63	0.5000	0.2333	0.2640	0.5135	0.4167
64	0.9000	0.7000	0.5134	0.7063	0.6078
65	0.5000	0.3000	0.2302	0.5255	0.2812
66	0.7000	0.4000	0.2963	0.6869	0.3526
67	0.0000	0.0333	0.0740	0.3727	0.0417
68	0.6000	0.6667	0.4212	0.6712	0.4429
69	0.2000	0.1000	0.0359	0.2994	0.0500
70	0.4000	0.4333	0.3263	0.6111	0.3385
71	0.9000	0.6667	0.4615	0.7645	0.4965
72	0.0000	0.0667	0.0978	0.4295	0.1268
73	1.0000	0.8000	0.4389	0.7388	0.4531
74	0.3000	0.3000	0.2362	0.5698	0.3467
75	0.4000	0.4667	0.4245	0.7584	0.5260
77	0.1000	0.1333	0.0548	0.2256	0.1731
78	0.4000	0.5333	0.2350	0.5580	0.3311
79	0.6000	0.4667	0.2423	0.6063	0.3537
80	0.5000	0.2667	0.3307	0.6591	0.4211
81	0.1000	0.2667	0.2290	0.5177	0.3037
82	0.4000	0.4000	0.2697	0.6102	0.4149
83	0.4000	0.4333	0.3508	0.6655	0.4444
84	0.7000	0.6333	0.4689	0.7562	0.4643
85	0.0000	0.0000	0.0009	0.0380	0.0167
86	0.8000	0.6667	0.7309	0.8919	0.7600
87	1.0000	0.7667	0.3689	0.5689	0.4133
88	1.0000	0.9333	0.5859	0.8004	0.6066
89	0.0000	0.0000	0.0450	0.3223	0.0000
90	0.5000	0.3667	0.1925	0.5897	0.2333
91	0.5000	0.1667	0.2110	0.5437	0.2174
92	0.5000	0.3667	0.2397	0.4960	0.2955

---

Continued on next page

Q	P10	P30	MAP	NDCG	Rprec
93	0.1000	0.1667	0.0877	0.4009	0.1351
94	0.5000	0.3333	0.1915	0.4720	0.2273
95	1.0000	0.8000	0.4535	0.6911	0.5435
96	0.1000	0.2333	0.1664	0.5931	0.1447
97	0.6000	0.4333	0.1397	0.3534	0.2556
98	0.9000	0.6667	0.3100	0.5678	0.3284
99	0.2000	0.2333	0.0732	0.3089	0.1757
100	0.3000	0.1333	0.2510	0.4957	0.2308
101	0.5000	0.3000	0.3038	0.6546	0.2941
102	0.1000	0.1667	0.0614	0.2133	0.1250
103	0.6000	0.6667	0.6037	0.6443	0.7308
104	0.5000	0.5000	0.3227	0.6361	0.4018
105	0.0000	0.0333	0.0261	0.2820	0.0000
106	0.3000	0.5000	0.3520	0.6774	0.3846
107	0.1000	0.2333	0.1931	0.5080	0.1646
108	0.9000	0.7000	0.4671	0.7073	0.4933
109	0.3000	0.3333	0.2238	0.4901	0.4348
110	0.2000	0.1333	0.0790	0.3292	0.1111
111	0.3000	0.2333	0.0589	0.1776	0.1528
112	0.7000	0.4000	0.4124	0.7866	0.4762
113	0.8000	0.2667	0.2194	0.4041	0.2500
114	0.4000	0.1667	0.1689	0.4529	0.1739
115	0.3000	0.3667	0.2288	0.5300	0.3115
116	1.0000	0.7667	0.3367	0.6689	0.4031
117	0.6000	0.3000	0.3153	0.7554	0.3000
118	0.9000	0.9000	0.5381	0.6571	0.5720
119	1.0000	0.5000	0.4717	0.7657	0.4054
120	1.0000	0.7667	0.5100	0.7288	0.4902
121	0.9000	0.9000	0.6356	0.7917	0.6462
122	0.0000	0.0000	0.0000	0.0000	0.0000
123	0.9000	0.9000	0.2605	0.4466	0.3216
124	0.6000	0.3667	0.1529	0.4717	0.2847
125	1.0000	0.7000	0.2882	0.6843	0.3358
126	0.9000	0.9333	0.3411	0.5988	0.4087
127	1.0000	1.0000	0.6988	0.7608	0.6719
128	1.0000	0.8333	0.3507	0.5769	0.4112
129	1.0000	1.0000	0.8514	0.9484	0.7586

Continued on next page

---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
130	0.4000	0.2000	0.1266	0.4913	0.2353
131	0.7000	0.4000	0.4888	0.6554	0.5000
132	0.0000	0.1000	0.0562	0.2564	0.1406
133	1.0000	1.0000	0.3727	0.5754	0.4650
134	0.0000	0.0000	0.0028	0.0610	0.0172
135	1.0000	0.5000	0.4027	0.7586	0.4706
136	0.2000	0.1000	0.0071	0.0458	0.0286
137	0.3000	0.3333	0.1107	0.4088	0.1812
138	1.0000	1.0000	0.7397	0.9126	0.7039
139	1.0000	0.8000	0.5354	0.8226	0.5185
140	0.9000	0.6667	0.2343	0.5761	0.2482
141	1.0000	0.9667	0.6012	0.6740	0.6283
142	0.9000	0.3333	0.2670	0.6048	0.3711
143	0.9000	0.6000	0.6067	0.7357	0.6190
144	0.9000	0.7333	0.4541	0.6881	0.5244
145	0.9000	0.7000	0.3151	0.6986	0.3023
146	1.0000	1.0000	0.6455	0.7201	0.6218
147	0.2000	0.5000	0.4518	0.6916	0.5857
148	0.6000	0.2333	0.1277	0.4836	0.1633
149	1.0000	0.6667	0.4695	0.7870	0.5125
150	0.0000	0.0000	0.0004	0.0154	0.0000
151	0.0000	0.0000	0.0000	0.0000	0.0000
152	1.0000	0.7667	0.7637	0.8554	0.8214
153	0.9000	0.4667	0.3177	0.7427	0.3049
154	0.4000	0.5000	0.3614	0.5950	0.4595
155	0.3000	0.2333	0.2181	0.5324	0.2727
156	1.0000	0.9333	0.8300	0.8906	0.7979
157	1.0000	1.0000	0.5085	0.6800	0.5134
158	0.0000	0.0667	0.1794	0.4812	0.2133
159	0.8000	0.6333	0.7661	0.8956	0.8182
160	0.0000	0.0000	0.0199	0.2402	0.0000
161	0.4000	0.6667	0.2651	0.5868	0.2997
162	0.6000	0.6000	0.3980	0.6622	0.4762
163	1.0000	1.0000	0.6909	0.8290	0.6682
164	0.8000	0.6333	0.3564	0.6970	0.3048
165	0.1000	0.0333	0.1436	0.3067	0.1429
166	0.8000	0.4000	0.8043	0.9367	0.7692

---

Continued on next page

Q	P10	P30	MAP	NDCG	Rprec
167	0.3000	0.2000	0.0806	0.3704	0.1825
168	0.3000	0.4333	0.2423	0.5573	0.2647
169	1.0000	1.0000	0.8820	0.9734	0.8727
170	1.0000	0.8667	0.4738	0.8130	0.4531
171	0.7000	0.8667	0.5887	0.8845	0.4896
172	1.0000	1.0000	0.8533	0.9593	0.8110
173	0.0000	0.0000	0.0008	0.0348	0.0000
174	0.8000	0.3000	0.6299	0.8864	0.5333
175	0.3000	0.5333	0.3071	0.6345	0.3821
176	0.3000	0.1000	0.0099	0.1025	0.0312
177	0.9000	0.9667	0.7475	0.9295	0.7014
178	1.0000	0.9333	0.6621	0.8500	0.6530
179	0.1000	0.1000	0.2325	0.6106	0.1000
180	0.5000	0.8333	0.4363	0.6018	0.5182
181	1.0000	0.7000	0.7241	0.8703	0.6897
182	1.0000	0.8667	0.7729	0.9341	0.7039
183	1.0000	0.9667	0.8818	0.9633	0.8542
184	1.0000	0.9333	0.4597	0.8381	0.4142
185	1.0000	0.6667	0.9299	0.9696	0.8636
186	1.0000	0.7000	0.7383	0.9322	0.6111
187	1.0000	0.8667	0.8651	0.8541	0.8293
188	1.0000	0.6667	0.8254	0.9591	0.6286
189	0.0000	0.0000	0.0000	0.0056	0.0000
190	1.0000	1.0000	0.9009	0.9693	0.8507
191	1.0000	1.0000	0.7914	0.9300	0.7368
192	0.9000	0.8333	0.5961	0.6504	0.6364
193	1.0000	0.9667	0.6105	0.6460	0.5833
194	0.0000	0.2000	0.1307	0.4555	0.0625
195	1.0000	0.8333	0.4760	0.6888	0.5162
196	0.9000	0.9333	0.6180	0.8561	0.5868
197	0.9000	0.9000	0.5979	0.8374	0.5825
198	0.7000	0.7333	0.4332	0.5976	0.5224
199	1.0000	0.9667	0.8981	0.9676	0.8112
200	0.9000	0.7333	0.4644	0.8191	0.4896
201	1.0000	0.9000	0.7834	0.8922	0.7394
202	1.0000	1.0000	0.8923	0.9739	0.8714
203	0.5000	0.3333	0.1184	0.3650	0.2078

Continued on next page



---

<b>Q</b>	<b>P10</b>	<b>P30</b>	<b>MAP</b>	<b>NDCG</b>	<b>Rprec</b>
204	0.9000	0.7333	0.6655	0.8112	0.7097
205	1.0000	0.9667	0.6415	0.8905	0.6652
206	1.0000	0.9000	0.4159	0.8429	0.3636
207	1.0000	0.9000	0.7492	0.9029	0.6949
208	1.0000	0.9667	0.8742	0.9461	0.8137
209	1.0000	1.0000	0.5218	0.6899	0.5228
210	0.7000	0.5667	0.5172	0.8892	0.5312
211	0.8000	0.4000	0.4943	0.8353	0.5000
212	1.0000	0.6667	0.3874	0.8127	0.3577
213	1.0000	1.0000	0.9602	0.9928	0.9234
214	0.5000	0.5000	0.4924	0.7911	0.4706
215	0.9000	0.9000	0.5992	0.6205	0.6594
216	0.6000	0.3667	0.2135	0.6210	0.2706
217	0.9000	0.8000	0.2863	0.5924	0.3385
218	1.0000	0.9667	0.8265	0.9362	0.7665
219	0.3000	0.1000	0.0782	0.4162	0.1071
220	0.3000	0.1667	0.1234	0.3578	0.1786
221	1.0000	1.0000	0.8083	0.9355	0.7767
222	1.0000	0.9667	0.4447	0.5689	0.3757
223	0.1000	0.2000	0.0554	0.2908	0.1216
224	0.5000	0.3000	0.5089	0.8630	0.3846
225	0.9000	0.8667	0.7872	0.8506	0.7297

---



## References

---

- Abdul-Jaleel, N., Allan, J., Croft, W. B., Diaz, F., Larkey, L., Li, X., Smucker, M. D., and Wade, C. (2004). Umass at trec 2004: Novelty and hard. In *Proceedings of the TREC*. (page 71, 89, 117)
- Abel, F., Gao, Q., Houben, G. J., and Tao, K. (2011). Analyzing user modeling on twitter for personalized news recommendations. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*, pages 1–12. (page 51)
- Albakour, M.-D., Macdonald, C., and Ounis, I. (2013). On sparsity and drift for effective real-time filtering in microblogs. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 419–428. (page 9, 57)
- Albathan, M., Li, Y., and Algarni, A. (2013). Enhanced n-gram extraction using relevance feature discovery. In *Proceedings of the Australasian Joint Conference on Artificial Intelligence*, pages 453–465. (page 56)
- Albishre, K., Albathan, M., and Li, Y. (2015). Effective 20 newsgroups dataset cleaning. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 98–101. (page 18)

- Albishre, K., Li, Y., and Xu, Y. (2017). Effective pseudo-relevance for microblog retrieval. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW)*, pages 1–6. (page 18, 68)
- Albishre, K., Li, Y., and Xu, Y. (2018). Query-based automatic training set selection for microblog retrieval. In *Proceedings of the Advances in Knowledge Discovery and Data Mining - 22nd Pacific-Asia Conference (PAKDD)*, pages 325–336. (page 19, 84)
- Albishre, K., Li, Y., Xu, Y., and Huang, W. (2019). Query-based unsupervised learning for improving social media search. *World Wide Web*, pages 1–19. (page 19, 94)
- Amati, G., Amodeo, G., Bianchi, M., and Marcone, G. (2011). Fub, iasi-cnr, univaq at trec 2011. In *TREC*. (page 46)
- Amati, G. and Van Rijsbergen, C. J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4):357–389. (page 27)
- Andrzejewski, D. and Buttler, D. (2011). Latent topic feedback for information retrieval. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 600–608. (page 11)
- Andrzejewski, D., Zhu, X., and Craven, M. (2009). Incorporating domain knowledge into topic modeling via dirichlet forest priors. In *Proceedings of the 26th Annual International Conference on Machine Learning - ICML*. (page 63)
- Aslam, J. A. and Montague, M. (2001). Models for metasearch. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 276–284. (page 44)

- Baeza-Yates, R., Ribeiro, B. d. A. N., et al. (2011). *Modern information retrieval*. New York: ACM Press; Harlow, England: Addison-Wesley,. (page 33)
- Bandyopadhyay, A., Ghosh, K., Majumder, P., and Mitra, M. (2012). Query expansion for microblog retrieval. *International Journal of Web Science*, 1(4):368. (page 46, 56)
- Belkin, N. J. and Croft, W. B. (1992). Information filtering and information retrieval: two sides of the same coin? *Communications of the ACM*, 35(12):29–38. (page 30)
- Berendsen, R., Tsagkias, M., Weerkamp, W., and de Rijke, M. (2013). Pseudo test collections for training and tuning microblog rankers. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, page 53. (page 43, 57)
- Bernstein, M. S., Bakshy, E., Burke, M., and Karrer, B. (2013). Quantifying the invisible audience in social networks. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 21–30. (page 52)
- Bhowmik, A. and Ghosh, J. (2017). Letor methods for unsupervised rank aggregation. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1331–1340. (page 44)
- Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning - ICML*. (page 60)
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3(3):993–1022. (page 12, 59, 61, 117)
- Bouadjenek, M. R., Hacid, H., and Bouzeghoub, M. (2013a). Sopra: a new social personalized ranking function for improving web search. In *Proceedings of the 36th*

- international ACM SIGIR conference on Research and development in information retrieval*, pages 861–864. (page 52)
- Bouadjenek, M. R., Hacid, H., Bouzeghoub, M., and Daigremont, J. (2011). Personalized social query expansion using social bookmarking systems. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. (page 52)
- Bouadjenek, M. R., Hacid, H., Bouzeghoub, M., and Vakali, A. (2016). Persador: Personalized social document representation for improving web search. *Information Sciences*, 369:614–633. (page 52)
- Bouadjenek, M. R. M., Hacid, H., and Bouzeghoub, M. (2013b). Laicos: An open source platform for personalized social web search. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1446–1449. (page 52)
- Buckley, C., Salton, G., Allan, J., and Singhal, A. (1995). Automatic query expansion using smart : Trec 3. In *Proceedings of the TREC*. (page 8)
- Cai, J., Luo, J., Wang, S., and Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79. (page 10)
- Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. (2007). Learning to rank:from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136. (page 43)
- Caragiannis, I., Chatzigeorgiou, X., Krimpas, G. A., and Voudouris, A. A. (2019). Optimizing positional scoring rules for rank aggregation. *Artificial Intelligence*, 267:58–77. (page 44)

- Carmel, D. and Yom-Tov, E. (2010). Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 2(1):1–89. (page 82)
- Carpineto, C., de Mori, R., Romano, G., and Bigi, B. (2001). An information-theoretic approach to automatic query expansion. *ACM Transactions on Information Systems (TOIS)*, 19(1):1–27. (page 7)
- Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *Acm Computing Surveys (CSUR)*, 44(1):1–50. (page 7, 9, 29, 31, 56, 57)
- Cha, Y. and Cho, J. (2012). Social-network analysis using topic models. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 565–574. (page 60)
- Chakrabarti, D. and Punera, K. (2011). Event summarization using tweets. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*. (page 49)
- Chang, J., Boyd-Graber, J., Gerrish, S., Wang, C., and Blei, D. (2009). Reading tea leaves: How humans interpret topic models. In *Proceedings of the Advances in neural information processing systems*, pages 288–296. (page 61)
- Chang, Y., Dong, A., Kolari, P., Zhang, R., Inagaki, Y., Diaz, F., Zha, H., and Liu, Y. (2013). Improving recency ranking using twitter data. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):4. (page 39)
- Chen, J., Nairn, R., Nelson, L., Bernstein, M., and Chi, E. (2010). Short and tweet: experiments on recommending content from information streams. In *Proceedings*

- of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1185–1194. (page 82)
- Chen, K., Chen, T., Zheng, G., Jin, O., Yao, E., and Yu, Y. (2012). Collaborative personalized tweet recommendation. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 661–670. (page 51)
- Chen, Q., Hu, Q., Huang, J., and He, L. (2018). Taker: Fine-grained time-aware microblog search with kernel density estimation. *IEEE Transactions on Knowledge and Data Engineering*, 4347(c):1–1. (page 41, 57, 93)
- Chen, Y., Amiri, H., Li, Z., and Chua, T.-S. (2013a). Emerging topic detection for organizations from microblogs. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. (page 61, 62)
- Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., and Ghosh, R. (2013b). Discovering coherent topics using general knowledge. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. (page 64)
- Cheng, F., Zhang, X., He, B., Luo, T., and Wang, W. (2012). A survey of learning to rank for real-time twitter search. In *Proceedings of the Joint International Conference on Pervasive Computing and the Networked World*, pages 150–164. (page 42)
- Choi, J. and Croft, W. B. W. (2012). Temporal models for microblogs. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, page 2491. (page 41, 58)



- Chong, F. and Chua, T. (2013). Automatic summarization of events from social media. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media Automatic*. (page 49, 62)
- Chuang, J., Gupta, S., Manning, C., and Heer, J. (2013). Topic model diagnostics: Assessing domain relevance via topical alignment. In *Proceedings of the 30th International Conference on machine learning (ICML-13)*, pages 612–620. (page 118)
- Chy, A. N., Ullah, M. Z., and Aono, M. (2019). Query expansion for microblog retrieval focusing on an ensemble of features. *Journal of Information Processing*, 27:61–76. (page 57)
- Croft, W. B., Metzler, D., and Strohman, T. (2015). *Search Engines: Information Retrieval in Practice*. Pearson Education, Inc. (page 25, 27, 28)
- Damak, F., Pinel-Sauvagnat, K., Boughanem, M., and Cabanac, G. (2013). Effectiveness of state-of-the-art features for microblog search. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, pages 914–919. (page 39)
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. (page 154)
- Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., Zheng, Z., and Zha, H. (2010). Time is of the essence: improving recency ranking using twitter data. In *Proceedings of the 19th international conference on World wide web*, pages 331–340. (page 29, 58)
- Duan, Y., Chen, Z., Wei, F., Zhou, M., and Shum, H.-Y. (2012a). Twitter topic

- summarization by ranking tweets using social influence and content quality. In *Proceedings of COLING*, pages 763–780. (page 48)
- Duan, Y., Jiang, L., Qin, T., Zhou, M., and Shum, H.-Y. (2010). An empirical study on learning to rank of tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 295–303. (page 36, 40, 42)
- Duan, Y., Wei, F., Zhou, M., and Shum, H.-Y. (2012b). Graph-based collective classification for tweets. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2323–2326. (page 38)
- Efron, M. (2010a). Hashtag retrieval in a microblogging environment. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 787–788. (page 36)
- Efron, M. (2010b). Linear time series models for term weighting in information retrieval. *Journal of the American Society for Information Science and Technology*, 61(7):1299–1312. (page 41)
- Efron, M. (2011). Information search and retrieval in microblogs. *Journal of the American Society for Information Science and Technology*, 62(6):996–1008. (page 5, 36)
- Efron, M. and Golovchinsky, G. (2011). Estimation methods for ranking recent information. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 495–504. (page 58)
- Efron, M., Lin, J., He, J., and De Vries, A. (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*, pages 33–42. (page 41, 58, 116)

- Efron, M., Organisciak, P., and Fenlon, K. (2012). Improving retrieval of short texts through document expansion. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 911–920. (page 57)
- El-Ganainy, T., Magdy, W., and Rafea, A. (2014). Hyperlink-extended pseudo relevance feedback for improved microblog retrieval. In *Proceedings of the first international workshop on Social media retrieval and analysis*, pages 7–12. (page 57)
- Erkan, G. and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479. (page 47, 48)
- Fan, F., Feng, Y., Yao, L., and Zhao, D. (2016). Adaptive evolutionary filtering in real-time twitter stream. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1079–1088. (page 49)
- Fan, F., Qiang, R., Lv, C., and Yang, J. (2015). Improving microblog retrieval with feedback entity model. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 573–582. (page 9, 57)
- Fang, A., Ounis, I., Habel, P., and Macdonald, C. (2015). Topic-centric classification of twitter user’s political orientation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 791–794. (page 52)
- Feng, W. and Wang, J. (2013). Retweet or not?: personalized tweet re-ranking. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 577–586. (page 50)

- Ge, T., Cui, L., Chang, B., Sui, Z., and Zhou, M. (2016). Event detection with burst information networks. In *Proceedings of the 26th International Conference on Computational Linguistics*, pages 3276–3286. (page 29)
- Gheyas, I. A. and Smith, L. S. (2010). Feature subset selection in large dimensionality domains. *Pattern Recognition*, 43(1):5–13. (page 8)
- Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., and Schmidhuber, J. (2008). A novel connectionist system for unconstrained handwriting recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):855–868. (page 153)
- Han, Z., Li, X., Yang, M., Qi, H., Li, S., and Zhao, T. (2012). Hit at trec 2012 microblog track. In *Proceedings of the TREC*, pages 267–276. (page 46)
- Hasanain, M. and Elsayed, T. (2017). Query performance prediction for microblog search. *Information Processing and Management*, 53(6):1320–1341. (page 58)
- Hauff, C. (2010). Predicting the effectiveness of queries and retrieval systems. *SIGIR Forum*, 44(1):88. (page 13)
- He, B. and Ounis, I. (2009). Studying query expansion effectiveness. In *Proceedings of the European Conference on Information Retrieval*, pages 611–619. (page 13)
- He, R., Liu, Y., Yu, G., Tang, J., Hu, Q., and Dang, J. (2017). Twitter summarization with social-temporal context. *World Wide Web*, 20(2):267–290. (page 48)
- Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. (page 12, 59)

- Hong, L., Ahmed, A., Gurusurthy, S., Smola, A. J., and Tsioutsoulis, K. (2012). Discovering geographical topics in the twitter stream. In *Proceedings of the 21st international conference on World Wide Web*, pages 769–778. (page 39)
- Hong, L., Dan, O., and Davison, B. D. (2011). Predicting popular messages in twitter. In *Proceedings of the 20th international conference on World wide web*, pages 57–58. (page 61)
- Hong, L. and Davison, B. D. (2010). Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, pages 80–88. ACM. (page 62, 63)
- Inouye, D. and Kalita, J. K. (2011). Comparing twitter summarization algorithms for multiple post summaries. In *IEEE Third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computin*, pages 298–306. (page 48)
- Jagarlamudi, J., Daumé III, H., and Udupa, R. (2012). Incorporating lexical priors into topic models. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213. (page 61, 63)
- Järvelin, K. and Kekäläinen, J. (2002). Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446. (page 112)
- Java, A., Song, X., Finin, T., and Tseng, B. (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*, pages 56–65. (page 5)

- Jian, F., Huang, J. X., Zhao, J., He, T., and Hu, P. (2016). A simple enhancement for ad-hoc information retrieval via topic modelling. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 733–736. (page 11)
- Jin, O., Liu, N. N., Zhao, K., Yu, Y., and Yang, Q. (2011). Transferring topical knowledge from auxiliary long texts for short text clustering. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 775–784. (page 36, 63)
- Joachims, T. (2006). Training linear svms in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226. (page 42)
- Jones, R. and Diaz, F. (2007). Temporal profiles of queries. *ACM Transactions on Information Systems*, 25(3):14. (page 29, 40)
- Kanhabua, N., Blanco, R., Nørvåg, K., et al. (2015). Temporal information retrieval. *Foundations and Trends® in Information Retrieval*, 9(2):91–208. (page 29, 40)
- Kaplan, A. M. and Haenlein, M. (2011). The early bird catches the news: Nine things you should know about micro-blogging. *Business Horizons*, 54(2):105–113. (page 3)
- Kasisviswanathan, S. P., Melville, P., Banerjee, A., and Sindhvani, V. (2011). Emerging topic detection using dictionary learning. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 745–754. (page 55)
- Kedzie, C., McKeown, K., and Diaz, F. (2015). Predicting salient updates for disaster summarization. In *Proceedings of the 53rd Annual Meeting of the Association for*

- Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1608–1617. (page 30)
- Khudyak Kozorovitsky, A. and Kurland, O. (2011). Cluster-based fusion of retrieved lists. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 893–902. (page 44)
- Kleinberg, J. (2016). Temporal dynamics of on-line information streams. *Data Stream Management*, pages 221–238. (page 30)
- Kotov, A., Rakesh, V., Agichtein, E., and Reddy, C. K. (2015). Geographical latent variable models for microblog retrieval. In *Proceedings of the European Conference on Information Retrieval*, pages 635–647. (page 39)
- Kotov, A., Wang, Y., and Agichtein, E. (2013). Leveraging geographical metadata to improve search over social media. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 151–152. (page 39)
- Kulkarni, A., Teevan, J., Svore, K. M., and Dumais, S. T. (2011). Understanding temporal query dynamics. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 167–176. (page 29)
- Lancaster, F. W. and Fayen, E. G. (1973). *Information retrieval: on-line*. Melville Publishing Company, Los Angeles, California. (page 24)
- Laniado, D. and Mika, P. (2010). Making sense of twitter. In *Proceedings of the International Semantic Web Conference*, pages 470–485. (page 38)
- Lau, C. H., Li, Y., and Tjondronegoro, D. (2011). Microblog retrieval using topical features and query expansion. In *Proceedings of the TREC*. (page 7, 57)

- Lavrenko, V. and Croft, W. B. (2001). Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127. (page 10, 28, 32, 71, 89)
- Lee, K. S., Croft, W. B., and Allan, J. (2008). A cluster-based resampling method for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 235–242. ACM. (page 82)
- Lee, R. and Sumiya, K. (2010). Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL international workshop on location based social networks*, pages 1–10. (page 54)
- Li, C., Cheung, W. K., Ye, Y., Zhang, X., Chu, D., and Li, X. (2015a). The author-topic-community model for author interest profiling and community discovery. *Knowledge and Information Systems*, 44(2):359–383. (page 60)
- Li, M., Luo, L., Miao, L., Xue, Y., Zhao, Z., and Wang, Z. (2016). Friendrank: A personalized approach for tweets ranking in social networks. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM*, pages 896–900. (page 50)
- Li, P., Wang, Y., Gao, W., and Jiang, J. (2011). Generating aspect-oriented multi-document summarization with event-aspect model. In *Proceedings of the conference on empirical methods in Natural Language Processing*, pages 1137–1146. Association for Computational Linguistics. (page 50)
- Li, X. and Croft, W. B. (2003). Time-based language models. In *Proceedings of*



- the twelfth international conference on Information and knowledge management, CIKM*, pages 469–475. (page 29, 30, 58, 89, 116)
- Li, Y., Algarni, A., Albathan, M., Shen, Y., and Bijaksana, M. A. (2015b). Relevance feature discovery for text mining. *IEEE Transactions on Knowledge and Data Engineering*, 27(6):1656–1669. (page 56, 96, 109)
- Li, Y., Algarni, A., and Zhong, N. (2010). Mining positive and negative patterns for relevance feature discovery. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 753–762. (page 56)
- Li, Y., Zhou, X., Bruza, P., Xu, Y., and Lau, R. Y. (2012). A two-stage decision model for information filtering. *Decision Support Systems*, 52(3):706–716. (page 56)
- Liang, S. and de Rijke, M. (2015). Burst-aware data fusion for microblog search. *Information Processing and Management*, 51(2):89–113. (page 44, 58)
- Liang, S., De Rijke, M., and Tsagkias, M. (2013). Late data fusion for microblog search. In *Proceedings of the European Conference on Information Retrieval*, pages 743–746. (page 44)
- Liang, S., Ren, Z., and De Rijke, M. (2014a). Fusion helps diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 303–312. (page 44)
- Liang, S., Ren, Z., Weerkamp, W., Meij, E., and De Rijke, M. (2014b). Time-aware rank aggregation for microblog search. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 989–998. (page 44)

- Lin, C. C., Lin, C. C., Li, J., Wang, D., Chen, Y., and Li, T. (2012). Generating event storylines from microblogs. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 175–184. (page 9)
- Lin, J., Snow, R., and Morgan, W. (2011). Smoothing techniques for adaptive online language models: Topic tracking in tweet streams. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 422–429. (page 37, 63)
- Lin, T., Tian, W., Mei, Q., and Cheng, H. (2014). The dual-sparse topic model: mining focused topics and focused terms in short text. In *Proceedings of the 23rd international conference on World wide web*, pages 539–550. (page 12, 63)
- Lipizzi, C., Dessavre, D. G., Iandoli, L., and Marquez, J. E. R. (2016). Towards computational discourse analysis: A methodology for mining twitter backchanneling conversations. *Computers in Human Behavior*, 64:782–792. (page 49)
- Liu, F., Liu, Y., and Weng, F. (2011). Why is sxsw trending?: exploring multiple text sources for twitter topic summarization. In *Proceedings of the Workshop on Languages in Social Media*, pages 66–75. (page 49, 50)
- Liu, K.-L., Li, W.-J., and Guo, M. (2012a). Emoticon smoothed language models for twitter sentiment analysis. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*. (page 53)
- Liu, T.-Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331. (page 42)
- Liu, X., Li, Y., Wei, F., and Zhou, M. (2012b). Graph-based multi-tweet

- summarization using social signals. In *Proceedings of COLING*, pages 1699–1714. (page 48)
- Liu, Y.-T., Liu, T.-Y., Qin, T., Ma, Z.-M., and Li, H. (2007). Supervised rank aggregation. In *Proceedings of the 16th international conference on World Wide Web*, pages 481–490. (page 44)
- Long, R., Wang, H., Chen, Y., Jin, O., and Yu, Y. (2011). Towards effective event detection, tracking and summarization on microblog data. In *Proceedings of the International Conference on Web-Age Information Management*, pages 652–663. (page 55)
- Losada, D. E., Parapar, J., and Barreiro, A. (2018). A rank fusion approach based on score distributions for prioritizing relevance assessments in information retrieval evaluation. *Information Fusion*, 39:56–71. (page 44)
- Louvan, S., Ibrahim, M., and Adriani, M. (2011). University of indonesia at trec 2011 microblog track. In *TREC*. (page 46)
- Luo, Z., Osborne, M., Petrovic, S., and Wang, T. (2012). Improving twitter retrieval by exploiting structural information. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 648–654. (page 40)
- Lv, C., Qiang, R., Fan, F., and Yang, J. (2015). Knowledge-based query expansion in real-time microblog search. In *Proceedings of the Asia Information Retrieval Symposium*, pages 43–55. (page 116)
- Lv, Y. and Zhai, C. (2009a). Adaptive relevance feedback in information retrieval. In *Proceedings of the ACM conference on Conference on Information and Knowledge Management*, pages 255–264. (page 84)

- Lv, Y. and Zhai, C. (2009b). A comparative study of methods for estimating query language models with pseudo feedback. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1895–1898. ACM. (page 71)
- Lv, Y. and Zhai, C. (2010). Positional relevance model for pseudo-relevance feedback. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 579–586. ACM. (page 32, 82)
- Lv Chao Fan, F. Q. R. F. Y. and Yang, J. (2014). Pkuicst at trec 2014 microblog track: Feature extraction for effective microblog search and adaptive clustering algorithms for ttg. In *Proceedings of the TREC*. (page 143)
- Lynam, T. R., Buckley, C., Clarke, C. L., and Cormack, G. V. (2004). A multi-system analysis of document and term selection for blind feedback. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 261–269. ACM. (page 8)
- Manning, C., Raghavan, P., and Schütze, H. (2008). *Introduction to information retrieval*, volume 1. Cambridge university press. (page 22)
- Martins, F. and Callan, J. (2018). A vertical prf architecture for microblog search. In *Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 107–114. (page 57)
- Martins, F., Magalhães, J., and Callan, J. (2016). 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*, pages 667–676. (page 57)
- Massoudi, K., Tsagkias, M., De Rijke, M., and Weerkamp, W. (2011). Incorporating

- query expansion and quality indicators in searching microblog posts. *Advances in information retrieval*, pages 362–367. (page 56)
- Mathioudakis, M. and Koudas, N. (2010). Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 115–1158. (page 54)
- McCay-Peet, L. and Quan-Haase, A. (2017). What is social media and what questions can social media research help us answer. *The SAGE handbook of social media research methods*, pages 13–26. (page 33)
- McCreadie, R. and Macdonald, C. (2013). Relevance in microblogs: Enhancing tweet retrieval using hyperlinked documents. In *Proceedings of the 10th conference on open research areas in information retrieval*, pages 189–196. (page 36, 40)
- McCreadie, R., Macdonald, C., and Ounis, I. (2014). Incremental update summarization: Adaptive sentence selection based on prevalence and novelty. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 301–310. (page 30)
- McDonald, R. (2007). A study of global inference algorithms in multi-document summarization. In *Proceedings of the European Conference on Information Retrieval*, pages 557–564. (page 50)
- Mehrotra, R., Sanner, S., Buntine, W., and Xie, L. (2013). Improving lda topic models for microblogs via tweet pooling and automatic labeling. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 889–892. (page 62)
- Meng, X., Wei, F., Liu, X., Zhou, M., Li, S., and Wang, H. (2012). Entity-centric topic-oriented opinion summarization in twitter. In *Proceedings of the 18th ACM*

- SIGKDD international conference on Knowledge discovery and data mining*, pages 379–387. (page 53)
- Metzler, D., Cai, C., and Hovy, E. (2012). Structured event retrieval over microblog archives. In *Proceeding NAACL HLT '12 Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 646–655. (page 55, 57)
- Metzler, D., Cai, C., and Rey, M. (2011). Use / isi at trec 2011 : Microblog track. In *Proceedings of the TREC*. (page 46)
- Metzler, D., Jones, R., Peng, F., and Zhang, R. (2009). Improving search relevance for implicitly temporal queries. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 700–701. (page 29)
- Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*. (page 48)
- Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*. (page 153)
- Miyanishi, T., Seki, K., and Uehara, K. (2013). Improving pseudo-relevance feedback via tweet selection. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 439–448. (page 9, 56, 57, 82, 93)
- Miyanishi, T., Seki, K., and Uehara, K. (2014). Time-aware latent concept expansion for microblog search. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, pages 366–375. (page 57)

- Nagmoti, R., Teredesai, A., and De Cock, M. (2010). Ranking approaches for microblog search. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, pages 153–157. (page 38, 40)
- Nenkova, A., McKeown, K., et al. (2011). Automatic summarization. *Foundations and Trends® in Information Retrieval*, 5(2):103–233. (page 46)
- Nguyen, M. and Nguyen, M. (2017). Intra-relation or inter-relation?: exploiting social information for web document summarization. *Expert Systems with Applications*, 76:71–84. (page 47)
- Nguyen, M.-T., Tran, D.-V., and Nguyen, L.-M. (2018). Social context summarization using user-generated content and third-party sources. *Knowledge-Based Systems*, 144:51–64. (page 47)
- Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Machine learning*, 39(2-3):103–134. (page 62)
- O'Connor, B., Balasubramanyan, R., Routledge, B. R., and Smith, N. A. (2010). From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the Fourth International AAI Conference on Weblogs and Social Media*. (page 52)
- Ounis, I., Macdonald, C., Lin, J., and Soboroff, I. (2011). Overview of the trec-2011 microblog track. In *Proceedings of the TREC*, volume 32. (page 7, 46, 56, 106, 111, 141, 143)
- Petrovi, S., Osborne, M., and Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Proceedings of the Annual Conference of the North*

- American Chapter of the Association for Computational Linguistics*, pages 181–189. (page 54)
- Phan, X.-H., Nguyen, L.-M., and Horiguchi, S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceeding of the 17th international conference on World Wide Web*, pages 91–100. (page 62)
- Ponte, J. M. and Croft, W. B. (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*, pages 275–281, New York, New York, USA. ACM. (page 27, 69)
- Popescu, A.-M. and Pennacchiotti, M. (2010). Detecting controversial events from twitter. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1873–1876. (page 52)
- Porteous, I., Newman, D., Ihler, A., Asuncion, A., Smyth, P., and Welling, M. (2008). Fast collapsed gibbs sampling for latent dirichlet allocation. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 569–577. (page 69)
- Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137. (page 110)
- Qian, M. and Zhai, C. (2013). Robust unsupervised feature selection. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. (page 10, 32)
- Qiang, R., Liang, F., and Yang, J. (2013). Exploiting ranking factorization machines



- for microblog retrieval. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 1783–1788. (page 43)
- Ravikumar, S., Talamadupula, K., Balakrishnan, R., and Kambhampati, S. (2013). Raprop: Ranking tweets by exploiting the tweet/user/web ecosystem and inter-tweet agreement. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2345–2350. (page 39)
- Ren, Z., Inel, O., Aroyo, L., and de Rijke, M. (2016). Time-aware multi-viewpoint summarization of multilingual social text streams. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 387–396. (page 53)
- Ren, Z., Liang, S., Meij, E., and de Rijke, M. (2013). Personalized time-aware tweets summarization. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 513–522. (page 51)
- Rendle, S. (2012). Learning recommender systems with adaptive regularization. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 133–142. (page 43)
- Robertson, S. (1977). The probability ranking principle in ir. *Journal of Documentation*, 33:294–304. (page 27)
- Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., and Gatford, M. (1994). Okapi at trec-3. In *Proceedings of 3rd Text REtrieval Conference*, pages 109–126. (page 10, 27, 96, 115)
- Rodriguez Perez, J. A. and Jose, J. M. (2015). On microblog dimensionality

- and informativeness: Exploiting microblogs' structure and dimensions for ad-hoc retrieval. In *Proceedings of the International Conference on The Theory of Information Retrieval*, pages 211–220. (page 7)
- Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. (page 60)
- Saha, A. and Sindhvani, V. (2012). Learning evolving and emerging topics in social media: A dynamic nmf approach with temporal regularization. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 693–702. (page 55)
- Sahami, M. and Heilman, T. D. (2006). A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web*, pages 377–386. (page 63)
- Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. (page 4, 54)
- Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620. (page 25)
- Sanderson, M. and Zobel, J. (2005). Information retrieval system evaluation: effort, sensitivity, and reliability. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 162–169. (page 114)
- Serizawa, M. and Kobayashi, I. (2013). A study on query expansion based on topic distributions of retrieved documents. In *Proceedings of the International*

- Conference on Intelligent Text Processing and Computational Linguistics*, pages 369–379. (page 11)
- Sharifi, B., Hutton, M.-A., and Kalita, J. K. (2010). Experiments in microblog summarization. In *Proceedings of the International Conference on Social Computing*, pages 49–56. (page 37, 49)
- Shaw, J. A. and Fox, E. A. (1994). Combination of multiple searches. In *Proceedings of the TREC*. (page 44)
- Sheldon, D., Shokouhi, M., Szummer, M., and Craswell, N. (2011). Lambdamerge: merging the results of query reformulations. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 795–804. (page 44)
- Shi, T., Kang, K., Choo, J., and Reddy, C. K. (2018). Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1105–1114. (page 117)
- Shou, L., Wang, Z., Chen, K., and Chen, G. (2013). Sumblr : Continuous summarization of evolving tweet streams. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 533–542. (page 49)
- Smucker, M. D., Allan, J., and Carterette, B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 623–632. (page 115)
- Soboroff, I., Ounis, I., and Lin, J. (2013). Overview of the trec-2013 microblog track. In *Proceedings of the TREC*. (page 7, 56, 106, 111, 141, 143)

- Soboroff, I., Ounis, I., and Lin, J. (2014). Overview of the trec-2014 microblog track. In *Proceedings of the TREC*. (page 7, 111, 141)
- Soboroff, I., Ounis, I., Macdonald, C., and Lin, J. (2012). Overview of the trec-2012 microblog track. In *Proceedings of the TREC*. (page 7, 46, 56, 111, 141)
- Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21. (page 26)
- Sriram, B., Fuhry, D., Demir, E., Ferhatosmanoglu, H., and Demirbas, M. (2010). Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842. (page 11)
- Tan, L., Roegiest, A., Clarke, C. L., and Lin, J. (2016). Simple dynamic emission strategies for microblog filtering. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 1009–1012. (page 58)
- Tang, J., Meng, Z., Nguyen, X., Mei, Q., and Zhang, M. (2014). Understanding the limiting factors of topic modeling via posterior contraction analysis. In *Proceedings of the International Conference on Machine Learning*, pages 190–198. (page 61)
- Tao, K., Abel, F., and Hauff, C. (2013). Groundhog day: near-duplicate detection on twitter. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1273–1283. (page 36)
- Teevan, J., Ramage, D., and Morris, M. (2011). # twittersearch: a comparison of microblog search and web search. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 35–44. (page 6, 40)

- Thonet, T., Cabanac, G., Boughanem, M., and Pinel-Sauvagnat, K. (2017). Users are known by the company they keep: Topic models for viewpoint discovery in social networks. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 87–96. (page 53)
- Van Rijsbergen, C. (1979). Information retrieval: theory and practice. In *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, pages 1–14. (page 24)
- Volkovich, Y. and Kaltenbrunner, A. (2011). Evaluation of valuable user generated content on social news web sites. In *Proceedings of the 20th international conference companion on World wide web*, pages 139–140. (page 33)
- Wallach, H. M., Mimno, D. M., and McCallum, A. (2009). Rethinking lda : Why priors matter. In *Proceedings of the Advances in neural information processing systems*, pages 1973–1981. (page 63)
- Wang, S., Lu, K., Lu, X., and Wang, B. (2014). Query dependent time-sensitive ranking model for microblog search. In *Proceedings of the Asia-Pacific Web Conference*, pages 644–651. (page 43)
- Wang, X., Fang, H., and Zhai, C. (2008). A study of methods for negative relevance feedback. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, page 219. (page 10)
- Wang, X. and McCallum, A. (2006). Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 424–433. (page 60, 61)
- Wang, Y., Huang, H., and Feng, C. (2017). Query expansion based on a feedback

- concept model for microblog retrieval. In *Proceedings of the 26th International Conference on World Wide Web*, pages 559–568. (page 9, 57, 93)
- Wei, X. and Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, volume pages, page 178. (page 78)
- Wei, Z. and Gao, W. (2014). Utilizing microblogs for automatic news highlights extraction. In *Proceedings of the 25th International Conference on Computational Linguistics:*, pages 277–296. (page 47)
- Wei, Z. and Gao, W. (2015). Gibberish, assistant, or master?: Using tweets linking to news for extractive single-document summarization. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1003–1006. (page 47)
- Weng, J., Lim, E.-P., Jiang, J., and He, Q. (2010). Twitterank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270. (page 62)
- Whiting, S., Klampanos, I. A., and Jose, J. M. (2012). Temporal pseudo-relevance feedback in microblog retrieval. In *Proceedings of the European Conference on Information Retrieval*, pages 522–526. (page 7)
- Willett, P. (2006). The porter stemming algorithm: Then and now. *Electronic Library and information Systems*, 40:219–223. (page 110)
- Wu, Q., Burges, C. J., Svore, K. M., and Gao, J. (2010). Adapting boosting for information retrieval measures. *Information Retrieval*, 13(3):254–270. (page 43)

- Xu, J. and Croft, W. B. (1996). Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. (page 32)
- Yan, X., Guo, J., Lan, Y., and Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1445–1456. (page 12, 62)
- Yang, Z., Cai, K., Tang, J., Zhang, L., Su, Z., and Li, J. (2011). Social context summarization. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 255–264. (page 47)
- Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764. (page 153)
- Ye, Z., Huang, J. X., and Lin, H. (2013). Finding a good query-related topic for boosting pseudo-relevance feedback. *International Review of Research in Open and Distance Learning*, 14(4):90–103. (page 14)
- Yi, X. and Allan, J. (2009). A comparative study of utilizing topic models for information retrieval. In *Proceedings of the European conference on information retrieval*, pages 29–41. (page 11)
- Zhai, C. and Lafferty, J. (2001a). Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 403–410. (page 32)
- Zhai, C. and Lafferty, J. (2001b). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual*

- international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM. (page 32, 96)
- Zhai, C. and Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*. (page 70)
- Zhai, C. and Massung, S. (2016). *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*. Association for Computing Machinery and Morgan; Claypool, New York, NY, USA. (page 10, 27, 28, 32, 70, 83, 114)
- Zhang, X., Chen, X., Chen, Y., Wang, S., Li, Z., and Xia, J. (2015). Event detection and popularity prediction in microblogging. *Neurocomputing*, 149:1469 – 1480. (page 30)
- Zhao, F., Zhu, Y., Jin, H., and Yang, L. T. (2016a). A personalized hashtag recommendation approach using lda-based topic model in microblog environment. *Future Generation Computer Systems*, 65:196–206. (page 52)
- Zhao, W. X., Jiang, J., Weng, J., He, J., and Lim, E.-p. (2011). Comparing twitter and traditional media using topic models. In *Proceedings of the European Conference on Information Retrieval*, pages 338–349. (page 61, 62)
- Zhao, Y., Liang, S., and Ma, J. (2016b). Personalized re-ranking of tweets. In *Proceedings of the International Conference on Web Information Systems Engineering*, pages 70–84. (page 50)
- Zhong, N., Li, Y., and Wu, S.-t. (2010). Effective Pattern Discovery for Text Mining. *IEEE Transactions on Knowledge and Data Engineering*, 24(1):1–36. (page 96)



- Zhou, D., Lawless, S., and Wade, V. (2012). Improving search via personalized query expansion using social media. *Information Retrieval*, 15(3-4):218–242. (page 52)
- Zhu, B., Gao, J., Han, X., Shi, C., Liu, S., Liu, Y., and Cheng, X. (2012). Ictnet at microblog track trec 2012. In *Proceedings of the TREC*. (page 46)
- Zhu, X., Huang, J., Zhou, B., Li, A., and Jia, Y. (2017). Real-time personalized twitter search based on semantic expansion and quality model. *Neurocomputing*, 254:13–21. (page 51)
- Zingla, M. A., Chiraz, L., and Slimani, Y. (2016). Short query expansion for microblog retrieval. *Procedia Computer Science*, 96:225–234. (page 57)
- Zuo, Y., Wu, J., Zhang, H., Lin, H., Wang, F., Xu, K., and Xiong, H. (2016). Topic modeling of short texts: A pseudo-document view. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2105–2114. (page 117)





