Time-Aware Information Retrieval in Social Networks

Miyanishi, Taiki

(Degree)
博士（工学）

(Date of Degree)
2014-03-25

(Date of Publication)
2016-03-25

(Resource Type)
doctoral thesis

(Report Number)
甲第6106号

(URL)
https://hdl.handle.net/20.500.14094/D1006106

Doctoral Dissertation

Time-Aware Information Retrieval in Social Networks

ソーシャルネットワーク上の時間情報を考慮した情報検索

January 2014

Graduate School of System Informatics
Kobe University

Taiki Miyanishi

Doctoral Dissertation

Time-Aware Information Retrieval in Social Networks

ソーシャルネットワーク上の時間情報を考慮した情報検索

January 2014

Graduate School of System Informatics
Kobe University

Taiki Miyanishi

Abstract

Social network services are often used for the sharing of ideas or for communicating with other people. Such services can be valuable information sources. For example, one can immediately find information about breaking news and unexpected news events such as earthquakes or natural disasters through searching for documents in social networks. However, traditional information retrieval approaches fail to find such unique information because they cannot accommodate a temporal perspective.

Unlike common Web contents, social media often have real-time features by which many documents are posted by crowds of people when a notable event occurs. Moreover, the authorities of documents change over time. Because of these characteristics of social media, it is likely that relevance will have a temporal dimension. This dissertation presents specific examination of temporal properties, especially real-time features, in social networks and presents a description of time-aware information retrieval methods to retrieve time-stamped documents related to a user's information needs and to identify promising users who can produce high-quality documents in social networks.

We demonstrate empirically that time-aware information retrieval methods are consistently and significantly more effective than many current state-of-the-art retrieval methods. Our experiments use a corpus of time-stamped information: a document collection sampled from a famous social network service (Twitter) and a co-authorship network of arXiv(hep-th) during 1994–2003. Results show that time plays an important role in weighting words, concepts, and users to retrieve relevant and useful information.

謝辞

はじめに，神戸大学大学院システム情報学研究科計算科学専攻情報知能工学科計算知能講座において，本研究を行う機会を与えてくださり，ご多忙な身にも関わらず，ご指導とご助力を頂戴しました上原邦昭教授と関和広准教授に深く感謝致します．上原先生には，研究者として生き残っていくための方法を学び，関先生には，研究遂行の方法，論文の書き方，社会人としてのマナーについて教えて頂きました．また，研究者として未熟な私の論文を辛抱強く添削していただいたことについて，この場をお借りて御礼申し上げます．

また，私が Microsoft Research Asia にてインターンを行った際，指導員を担当して頂いた早稲田大学の酒井哲也准教授に感謝致します．酒井先生には，研究の進め方のみならず論文の書き方について懇切丁寧に指導して頂きました．羅志偉先生と有木康雄先生にはお忙しい中，本博士論文の主査と副査を努めて頂いたことに感謝致します．

博士後期課程２年次から特別研究員に採用して頂いた日本学術振興会に感謝致します．日本学術振興会からの経済的支援がなければ，私が博士の学生として研究することはできませんでした．

研究面で様々な手助けをしてくださいました日本学術振興会の海外特別研究員をなされている白浜公章氏に感謝致します．また，研究生活全般にわたり数々のご協力をして頂いた秘書，丸山陽子さんに感謝致します．

また常に研究室を活気に満ちた雰囲気にしてくださいました同講座の博士前期課程 2 年次の熊南昂司君，松村憲君，熊渕健二君，東山翔平君，藤川和樹君，渡邊結衣さん，博士前期課程 1 年次の福井聡君，北口沙也香さん，小林まなみさん，佐々木健太君，Si Nan さん，Jin Cheng 君，学部 4 年次の石川琢朗君，永井慶太郎君，折口卓巳君，川原駿君，田中優子さん，吉原輝君に感謝致します．また，同講座で互いに学び，励まし合い，苦労を分かち合った NTT コミュニケーション基礎科学研究所の研究員 Mathieu Blondel と同講座の博士後期課程 3 年次の Guo Xinlu さんにこの場を借りて感謝致します．

最後に，私の両親である宮西三枝，宮西繁樹に感謝致します．私の博士進学という我儘を快諾して下さり，ありがとうございました．

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Social network services (SNS) such as Twitter[1], Tumbler[2], and Facebook[3] have been developing rapidly. They now have a huge number of registered users. In SNS, especially microblog services, crowds of people post many documents when a notable event occurs [69]. For example, when an earthquake occurs, people often post documents that contain specific keywords such as "earthquake xxx area", "earthquake damage", and "xxx area evacuation center" to report and share useful information around them [33]. This is called real-time nature. Consequently, to find relevant and informative documents from such a large amount of time-stamped documents, information retrieval (IR) systems must incorporate a real-time quality into IR frameworks [19, 20, 22, 43, 44, 63]. We designate such frameworks as *time-aware information retrieval*.

However, such a time-aware IR method might retrieve unimportant documents because of a lack of document authority. In social networks, important and influential users tend to create high-quality documents. Their authority changes constantly over time. Consequently, this thesis presents a time-aware IR framework that simultaneously considers the real-time nature and user importance in social networks.

The remainder of this chapter is organized as follows. In Section 1.1 we present some challenges and motivations of information retrieval on social networks by analyzing documents from the social network and its structure. In Section 1.2 we state the main contributions of this dissertation. Finally, in Section 1.3 we provide an outline of the remainder of the dissertation.

[1]https://twitter.com/
[2]http://www.tumblr.com/
[3]https://www.facebook.com/

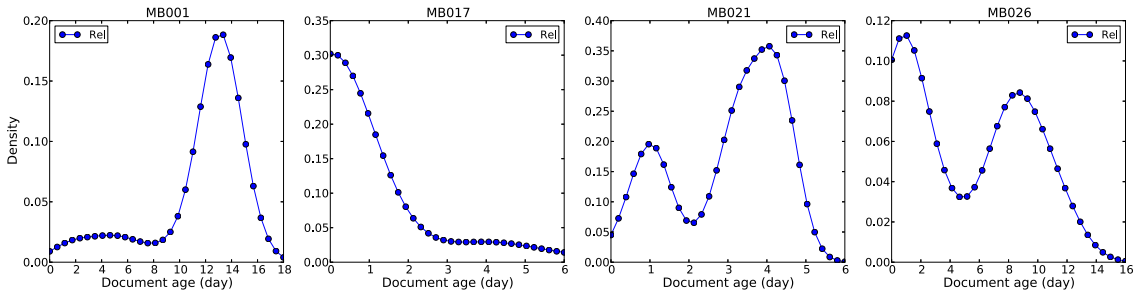

Figure 1.1: Temporal variations of four topics (MB001, MB017, MB021, and MB026.) from the TREC 2011 Microblog track based on relevant tweets. The x-axis shows the document age from the query time to the document time-stamp. The y-axis shows the kernel-estimated probability density for the document age. High density indicates the period during which the topic was described actively.

1.1 Challenge of Time-Aware Information Retrieval

In this section, we explain the motivation related to time-aware IR, and analyze time-stamped documents generated from a social network, especially Twitter, to demonstrate some challenges that social media present for current search engines.

One interesting property of social media is that many documents are posted by crowds of people when a notable event occurs. As a result, a set of documents related to the topic is an important clue for what topics are being described actively at a particular time. For example, when the news that "BBC World Service planned to close five of its language services"[4] was reported during January 25–27, 2011, many documents related to this event were posted actively at around this period. To inspect this feature in social networks, we used the Tweets2011 corpus[5], which consists of more than 16 million microblog documents called *tweets*, which are documents of a representative social network Twitter over a period of two weeks.

To clarify this temporal property of social media, we took four topics used in the TREC 2011 Microblog track [61]: "*BBC World Service staff cuts*" (MB001), "*White Stripes breakup*" (MB017), "*Emanuel residency court rulings*" (MB021), and "*US unemployment*" (MB026). The Kernel density estimates of the time-stamps of tweets relevant to these four topics are shown in Figure 1.1. Not all temporal variations of a given topic are the same. Moreover, many tweets are issued by users during the specified time period. Documents related to a given

[4]http://www.bbc.co.uk/news/entertainment-arts-12277413
[5]http://trec.nist.gov/data/tweets/

topic contain topic-related terms that appear frequently while the topic is being described. For example, tweets related to topic MB001 contain query terms: *BBC*, *cuts*, and *staff* as well as topic-related terms: *axe* and *jobs*. The important point is that if we were able to identify when a topic is being described actively, we would also be able to detect its related documents and related terms easily.

In addition to temporal variation, recency is an important temporal property. Some research has incorporated recency into microblog retrieval methods to search for recent and relevant tweets posted at around the time a query is issued [21, 50]. For example, a method considering recency is effective for retrieving tweets related to topics MB017 and MB026 in Figure 1.1, which exist almost entirely at around the query time. Furthermore, integrating recency into language modeling improves retrieval performance for retrieving documents posted in the recent past [20, 42]. These studies achieved great success in information retrieval. However, their models are insufficient for representing the temporal variation of a topic. For example, recency-based methods cannot accommodate specific temporal variations consisting of an old peak far from the query time or a multi-modal temporal variation (e.g., MB001 and MB021 in Figure 1.1). Consequently, they cannot discover terms that are temporally related to these topics. Other language model approaches incorporating temporal variation also performed well [16]; however, they can only ineffectively combine recency and the temporal variation of a topic in accordance with the type of temporal variation. In addition, these time-aware information retrieval methods mainly use word-based IR techniques under the assumption that query terms are mutually independent.

1.2 Contribution

In this section, we present a summary of the main contributions of this dissertation.

- We propose the time-aware query expansion framework, which leverages temporal properties (e.g., recency and temporal variation) derived from the real-time characteristic that many messages are posted by users when an interesting event has occurred recently.
- Additionally, we propose a simple method to combine the recency-based query expansion method and the temporal variation-based one according to the temporal variation of a given topic.
- We propose a concept-based query expansion method based on a temporal relevance model that uses the temporal variation of concepts (e.g. terms or

phrases) on microblogs. The proposed model produces important concepts that are used frequently within a particular time period associated with a given topic.

- We propose two-stage relevance feedback methods that apply document-dependent temporal pseudo-relevance feedback (PRF) to improved search results by an effective query expansion method with manual selection of a single relevant document to overcome the limitations of standard pseudo-relevance feedback.
- We empirically demonstrate that our two-stage relevance feedback approaches considerably and robustly improve search result relevance over almost all topics.
- We propose a framework to predict future significance or importance of users of a network through link prediction in a future network that changes naturally over time.
- The main emphasis of this dissertation is on improving the retrieval performance of time-stamped document retrieval in social networks. We demonstrate empirically that, for social media such as microblogs, time-aware information retrieval methods with consistently high effectiveness improve compared to other methods.

1.3 Dissertation Outline

The remainder of this dissertation is organized as follows.

- In Chapter 2, we survey the related work and provide some basic approaches for IR, which consist of language modeling frameworks for IR, existing time-aware IR methods, and network centrality to rank users in networks.
- In Chapter 3, we present word-based query expansion methods based on temporal relevance feedback to retrieve useful information from among many short documents being issued by crowds of people in social networks. The word-based temporal relevance model combines temporal properties (e.g., recency and temporal variation) according to the temporal variation of a given topic.
- In Chapter 4, we present a concept-based query expansion method based on a temporal relevance model that uses the temporal variation of concepts

on microblogs. The proposed model produces important concepts that are used frequently within a particular time period associated with a given topic, which have more power to facilitate discrimination between relevant and non-relevant microblog documents than words do.

- In Chapter 5, we present two-stage relevance feedback methods to overcome the limitation of a standard pseudo-relevance feedback method. To overcome the limitation of pseudo-relevance feedback for microblog searching, we propose a novel query expansion method based on two-stage relevance feedback that models search interests by manual tweet selection. Moreover, the proposed method integrates lexical and temporal evidence into its relevance model.

- In Chapter 6, we present a framework to predict future importance of users via link prediction. The proposed approach combines the results of link prediction and the importance of nodes by machine learning approaches.

- In Chapter 7, we summarize the findings of this dissertation and suggest directions for future research.

Chapter 2

Background and Related Work

In this chapter, in Section 2.1, we survey the related work on information retrieval techniques and object ranking methods. Then, in Section 2.2, we describe the theoretical framework for language modeling methods, temporal information retrieval techniques and basic object ranking measures. In addition, in Section 2.3, we describe the datasets we used for the evaluation. Finally, we conclude this chapter with Section 2.4

2.1 Related Work

2.1.1 Atemporal Information Retrieval

Standard Information Retrieval

Standard retrieval system mainly rely on the bag-of-words assumption that query terms are independently distributed in the set of documents. For example, Robertson and Jones [66] proposed the binary independence model that represents queries and documents as binary incidence vectors and ranks documents by their probability of relevance with respect to a given query. In addition, Robertson and Walker proposed BM25 [67] that uses term frequency as well as the document's length and collection statistics for scoring a document against a query. Ponte and Croft [64] proposed the query likelihood model assuming that the probability of a query is generated by the word probabilities on a document. Lavrenko and Croft [40] used language modeling framework for relevance feedback. Recently, Metzler and Crfot [53] combines the language modeling and inference network approaches into a single framework that is implemented on Indri search

engine [73].

Cluster Information Retrieval

Smoothing is a critical problem to improve the performance of language modeling frameworks [82, 83]. In order to smooth language model, many research use clustering information to rank documents. This framework is called cluster-based information retrieval [30, 37, 38, 47, 74, 77].

The cluster-based information retrieval mainly uses the topic-related cluster consisting of similar documents by using standard clustering algorithms such as k nearest neighbors, k-means and, Latent Dirichlet Allocation (LDA) [10]. Kurland and Lee [37] re-ranked documents using cluster information consisting of k nearest lexical similar documents. Liu and Croft [47] clustered all documents into several sets of similar documents using the k-means algorithm and used clusters for smoothing the language model with a global collection language model. Wei and Croft [77] proposed the document model using LDA to obtain cluster information for smoothing. Instead of smoothing for language models, Kalmanovich and Kurland [30] used cluster information with retrieved documents for creating an expanded query. In addition, Efron et al. [22] proposed a document expansion method based on the idea of Tao et al. [74], which smooths document language models by similar documents gathered with k nearest-neighbor. They submit documents as pseudo-queries to obtain similar documents, assuming that short documents tend to mention a single topic.

Concept-Based Information Retrieval

Modeling term dependencies is another direction to effectively retrieve relevant documents. It was reported recently that the concept-based IR method outperformed the word-based one across many tasks. Most successful works weight concept importance using a Markov Random Field (MRF), which generalizes unigram, bi-gram, and other various dependence models. The MRF models have improved retrieval performance significantly, especially for web search, where relevance at high ranks is particularly critical. For example, Metzler and Croft [54] proposed a query expansion method using the MRF model, which represents term-dependency for multiple terms (i.e. concepts) in a query. Moreover, they combine term dependence with query expansion using the MRF model, called Latent Concept Expansion (LCE) [55]. In fact, LCE outperformed a standard query expansion technique based on a bag-of-words model across several TREC datasets without decreasing search performance with regard to many queries. However,

LCE mainly use the document frequency on the importance of a query concept. It uses no concept information related to external sources. To overcome these shortcomings, Bendersky et al. [7] proposed a learning-to-rank approach for concept weighting, which uses internal and external sources such as Wikipedia, and a query log to obtain concept statistics. In addition, Bendersky et al. [8, 9] proposed learning-to-rank frameworks that weight concepts extracted from top retrieved documents by LCE as well as concepts in a query. Moreover, Bendersky and Croft [6] proposed the query formulation method which uses a combination of concepts represented by hyper-graphs generalizing term-dependencies. On both standard newswire and web TREC corpora, these concept-importance weighting approaches consistently and significantly outperform widely various state-of-the-art retrieval models.

However, these traditional word-based IR methods, cluster IR methods and concept weighting approaches do not consider temporal factors which, as described previously, are important factors for retrieving social media contents.

One of the goals of this dissertation is to address the issue of query term, concept, and documents weighting in a principled manner. In this dissertation we propose time-aware information retrieval frameworks to weight words, concepts, and documents.

2.1.2 Time-Aware Information Retrieval

With an increase of the number of time-stamped documents (e.g., news, weblogs, and microblogs), many studies have incorporated temporal properties into their respective frameworks. Dakka et al. [16] proposed a general ranking mechanism integrating temporal properties into a language model, thereby identifying the important periods for a given topic. Keikha et al. [32] proposed a time-based relevance model for improving blog retrieval. Moreover, Lin and Efron [44] reported that a temporal IR method for detecting topically related time significantly improves the microblog search performance. Massoudi et al. [50] proposed a QE method selecting terms temporally closer to the querytime. Using the notion of temporal profile [29], represented as a timeline for a set of documents returned by a search engine, Peetz et al. [63] proposed query modeling leveraging a temporal burst. Efron et al. [21] showed that the temporal variation of query and documents are keys to improve retrieval performance. Efron et al. [22] also proposed document expansion combining lexical and temporal information based on the notion of cluster IR.

Recency model was recently discussed in many works. Li and Croft [42] incorporated recency into the language model framework for IR [64]. to retrieve fresh

information such as news [40, 64]. Peetz et al. [62] tested many temporal document priors based on cognitive motivation for retrieving recent documents. Amati et al. [2] incorporated temporal recency into the document prior using survival function for microblog search. Massoudi et al. [50] proposed a query expansion method selecting words that are temporally closer to the query-time. Efron and Golovchinsky [20] proposed IR methods incorporating temporal properties, especially recency, into language modeling and showed their effectiveness for recency queries. Efron [19] also proposed a query-specific recency ranking approach.

Nevertheless, these existing time-aware IR methods mainly use word information and do not consider multi-term concepts and cluster information of documents. In addition, they did not incorporate document-dependent temporal variations into their query expansion model. Our method takes account of lexical evidence weighted by temporal evidence related to words, concepts, or documents.

2.1.3 Object Ranking in Social Networks

Social networks are dynamically changing. The detection of people who play a central role or influence other people in organizations and groups has been actively researched. Many researchers used the link-based object ranking which rank persons according to their surrounding relationships and developed the object ranking methods in various fields, for example Q&A forum [84], co-authorship network [46], Twitter [78], and so on. Most existing works focused on the importance and influence of objects in a network snapshot sliced at a given time and their algorithms can not be directly applied to a dynamic network which is continuing to change over time. In contrast, O'Madahain et all. [60] presented the method to rank the central objects on a dynamic network. Sayyadi et al. [70] proposed FutureRank that aims to predict future ranking of objects. O'Madahain recursively calculates the importance of objects according to the past and recent relationships; however, his approach can not predict the appearance of links in the future network. FutureRank sequentially predicts the importance of papers and authors in networks considering the documents' time-stamps. These object ranking methods predict object ranking in the future network; however, types of importance of objects they predict are limited. In this dissertation, we present the method which predict any importance of objects in the future network.

2.2 Fundamental Approach

In this section, we describe the proposed general theoretical framework for query representation and information retrieval based on language modeling approach. We start the section with Section 2.2.1, in which we formally describe the document ranking principle based on language modeling approach. Then, in Section 2.2.2, we introduce a basic framework of time-aware information retrieval. In Section 2.2.3, we present well-known object ranking measures called *network centrality*.

2.2.1 Language Model for Information Retrieval

Query Likelihood Model

Ponte and Croft [64] proposed simple but effective approach called query likelihood model. This model incorporates the assumption that the probability of a query Q is generated by the word probabilities on a document D. All documents are ranked in order of their probability of relevance or usefulness, which is defined as $P(D|Q)$. The posterior probability of a document $P(D|Q)$ by Bayes' rule becomes

$$P(D|Q) \propto P(Q|D)P(D), \tag{2.1}$$

where $P(Q|D)$ denotes the query likelihood on the given document and $P(D)$ stands for the prior probability that D is relevant to any query. To capture word frequency information in indexing a document, the multinomial model is used. This is called a uni-gram language model. We have the query likelihood $P(Q|D)$, where the query Q consists of n query terms $q_1, q_2, \ldots, q_n$, as

$$P(Q|D) = \prod_{i=1}^{n} P(q_i|D),$$

where $P(q_i|D)$ is the probability of a i-th query term q_i under the word distribution for document D. The maximum likelihood estimator of $P(q|D)$ is $P_{ml}(w|D) = \frac{f(w;D)}{\sum_{w' \in V} f(w';D)}$. Therein, $f(w;D)$ denotes the number of word counts of w in document D, $\sum_{w' \in V} f(w';D)$ is the number of words in D where V is the set of all words in the vocabulary. In most cases, this probability is applied to smoothing to temper over-fitting using a given collection. Among numerous smoothing methods, the following Dirichlet smoothing [83] is often used.

$$P(w|D) = \frac{|D|}{|D| + \mu} P_{ml}(w|D) + \frac{\mu}{|D| + \mu} P(w|C), \tag{2.2}$$

where μ is the Dirichlet prior and $P(w|C)$ is a uni-gram language model in a corpus C. Smoothing the maximum likelihood estimator of the uni-gram language model improves the estimated probabilities.

Word-Based Relevance Model

Lavrenko and Croft [40] incorporated relevance feedback into language modeling frameworks. They estimated a relevance model, $P(w|\mathcal{R})$, using a joint probability of observing the expanded word w together with query terms in query Q, assuming that the word w was sampled in the same way as the query terms from a distribution $\mathcal{R}$. That relevance model weights words w according to the following.

$$\begin{aligned} P(w|\mathcal{R}) \quad &\approx P(w|Q) = \sum_{D \in \mathcal{R}} P(w, D|Q) \\ &= \frac{1}{\mathcal{Z}} \sum_{D \in \mathcal{R}} P(w|D) P(Q|D) P(D) \\ &\propto \sum_{D \in \mathcal{R}} P(D) P(w|D) \prod_{i}^{n} P(q_i|D), \end{aligned} \tag{2.3}$$

where $\mathcal{R}$ is a set of relevant or pseudo-relevant document for query Q and where $\mathcal{Z} = \sum_{D \in \mathcal{R}} \sum_{w \in V} P(w, D, Q)$ is a normalization factor. When using the top M retrieved documents by the query Q for $\mathcal{R}$, this approach is called pseudo-relevance feedback. In addition, for query expansion, words w are ordered in descending order of $P(w|Q)$ in Equation (2.3). Then, the top k words are added to the original user query. Recall that this relevance model uses only word information.

Concept-based Relevance Model

To model query concepts through term dependencies for PRF, Metzler and Croft [55] proposed the concept-based PRF method called Latent concept expansion (LCE), which generates single and multi-term concepts that are related topically to an original query. These concepts are defined as *latent concepts*. To represent term-dependencies in a query and documents, LCE mainly uses the notion of Markov Random field [54]. Using LCE, users can automatically formulate the concepts a user has in mind, but which the user did not explicitly express in the query. The goal of LCE is to recover these latent concepts given some original query. As described in this paper, we used the simplified LCE proposed by Bendersky et al. [8] to assess the effectiveness of several components between baselines and our

proposed approach. Their LCE weights a latent concept extracted from pseudo-relevant documents $\mathcal{R}$ (top M retrieved documents) as follows:

$$S_{LCE}(c,Q) \propto \sum_{D \in \mathcal{R}} \exp\{\gamma_1 \phi_1(Q,D) + \gamma_2 \phi_2(c,D) - \gamma_3 \phi_3(c,C)\}, \tag{2.4}$$

where $\phi_1(Q,D)$ is a matching function between a document D and concepts in a query Q, $\phi_2(c,D)$ is a concept frequency in a document D, and $\phi_3(c,C)$ is a concept uni-gram $P(c|C)$ in the corpus C.

Moreover, we assume that the given query consisting of query concepts $c_1, c_2, \ldots, c_m$ in Q and the candidates of an expanded concept c in pseudo-relevant documents are sampled identically and independently from a concept uni-gram distribution of $\mathcal{R}$, namely, assuming the bag-of-concepts. When $\gamma_1 = \gamma_2 = \gamma_3 = 1$, $\phi_1(Q,D) = \log P(Q|D)$, $\phi_2(c,D) = \log P(c|D)$, $\phi_3(c,C) = 0$, we obtain the score function of a concept c in response to query Q as

$$S_{cRM}(c,Q) \propto \sum_{D \in \mathcal{R}} P(D)P(c|D) \prod_{i}^{m} P(\hat{q}_i|D), \tag{2.5}$$

where $\hat{q}_i$ is a i-th query concept in query Q. This PRF model drops the penalty of the inverse collection frequency of the concept in the corpus from Equation (2.4). In addition, the expansion of Equation (2.5) is similar to the word-based PRF model in Equation (2.3). Unlike the word-based PRF that uses only words, concept-based PRF in Equation (2.5) can use multi-term concepts as well as single words. However, existing word-based and concept-based methods can not use temporal information such as document time-stamps, which are important features for microblog searching.

2.2.2 Time-Based Language Model for Information Retrieval

Recency-Based Language Model

If we assume that the prior probability distribution over documents is uniform, then we rank documents in decreasing order of the query likelihood $P(Q|D)$ above. However, the quality of document is changing over time. Topically relevant but obsolete documents might not satisfy the user if recent information is preferred. Consequently, Li and Croft [42] incorporated a prior distribution considering recency over documents into language model frameworks for retrieval. They proposed application of the following exponential distribution as the document prior $P(D)$ to Equation (2.1). We have

$$P(D|t_D) = r \cdot e^{-r \cdot |t_Q - t_D|}, \tag{2.6}$$

where t_Q stands for the query time at which a query was issued by a user, t_D signifies a time-stamp of the document D, and r denotes a rate parameter of the exponential distribution. This model includes the assumption that newer documents have a higher probability than older ones do.

Recency-based relevance model

In addition, Li and Croft [42] incorporated recency into the relevance model redesigning the document prior as follows:

$$P(w|Q) \propto \sum_{D_i \in \mathcal{R}} P(D|t_D) P(w|D_i) \prod_{j}^{m} P(q_j|D_i), \tag{2.7}$$

where $P(D|t_D)$ denotes the recency-based document prior in Equation (2.6). This model is good at dealing with recency queries, but it is not able to accommodate any temporal variation. On microblog services, temporal dynamics of the topic varies, so that the recency-based method fails to find topic-related words having specific temporal variations consisting of an old peak that is distant from the query-time or a multi-modal temporal variation [57]. Furthermore, this model was not able to accommodate query-specific recency even though the degree of recency is topic-dependent [19, 20].

Time-Based Relevance Model

Keikha et al. [32] proposed a time-based relevance model. They assume that any topic relates to specific time and that their topic-related words are frequently used in this time. Their approach detects this topic-related time and incorporates this temporal property into language modeling frameworks as

$$P(w|Q) = \sum_{t} P(w|t,Q) P(t|Q). \tag{2.8}$$

The previous version by Choi and Croft [15] defined the word distribution $P(w|t,Q)$ at time t against a query Q as

$$P(w|t,Q) = \sum_{D_i \in \mathcal{R}_t} P(w|D_i) \prod_{j}^{m} P(q_j|D_i),$$

where $\mathcal{R}_t$ represents the top M documents issued in time t. Although the original work by Keikha et al. [32] assumed $P(w|t,Q)$ was uniform, Choi and Croft assumed that $P(w|t,Q)$ was equal to $P(w|Q)$ and incorporated the time property

into only $P(t|Q)$. This equation is the same to Equation (2.3) when using documents in time t except for $P(D)$ is set to be uniform, so that their model can consider word probability information in time t. Consequently, Equation (2.8) is interpreted as the weighted sum of $P(w|t,Q)$ by a temporal model $P(t|Q)$. The temporal model against a given query, $P(t|Q)$, is defined as

$$P(t|Q) = \frac{1}{Z} \sum_{D \in \mathcal{R}} P(t|D)P(Q|D), \tag{2.9}$$

where $P(t|D)$ is an indicator function $P(t|D) = 1$ if the date of t and a document time-stamp of D is the same; otherwise, $P(t|D) = 0$. One must recall that $P(Q|D)$ is the query likelihood of a document D for Q. Z is the normalization factor. It is particularly interesting that this definition is the same as the notion of the temporal profile proposed by Jones and Diaz [29]. This model estimates topic-related time using document time-stamps and search scores (i.e. query likelihoods assuming the prior probability of document $P(D)$ is uniform) of retrieved documents. This relevance model can weight the word distribution by this temporal profile, so it is able to capture general temporal variation by each topic. However, it ignores recency and document-dependent temporal information.

2.2.3 Network Centrality

The network centrality is the measure to quantify the influence or importance of objects in a network. This measure is defined by the structure of networks because, in general, the quality of objects highly depends on the links surrounding objects.

Overall, the following network centrality types are widely used.

- **Degree** [23]
  Degree centrality denotes how many objects are connected to a object in a network. It is defined as the number of objects adjoined to a given object through a direct link. In other words, it is defined as the number of links for $l_{ij} \in L$ given object i, where L is a set of links of the network.

- **Closeness** [23]
  Closeness centrality denotes how close a object is to all the other objects. It is defined as the average length of shortest paths of the object to all other objects. More formally, it is defined as the average of inverse distance d_{ij} between object i and object j ($j \in V, i \neq j$), $\frac{1}{\sum_{j \in V, j \neq i} \frac{d_{ij}}{n-1}}$, where V is a set of objects in networks.

- **Betweenness** [23]
  Betweenness centrality considers the objects bridging clusters to be important. It is calculated as the fraction of shortest paths $path_{jk}$ between object pairs j, k $(j, k \in V, j \neq i, k \neq i)$ that pass through the target object i.

- **PageRank** [11]
  PageRank denotes that they steadily visit on a certain object assuming random surfers who stochastically transit between two objects in a network, This centrality is based on the idea that an object linked to by many objects with high PageRank receives a high rank itself. PageRank is defined as $\vec{x} = \alpha \mathbf{P}^T + (1 - \alpha)\frac{\vec{1}}{n}$, where $\mathbf{P}$ is a transit matrix, n is the number of objects, and α is a parameter to control the ratio of teleportation.

When we consider users and their collaborations as objects and links in a network, **Degree** is the number of collaborators, **Closeness** is the closer relationship with other users through collaborations, **Betweenness** denotes the important users bridging different communities, Finally, **PageRank** denotes the popularity or the easiness to make collaborations in a community.

Note that these network centralities are used for the purpose of quantifying the importance of objects in stationary networks; however, social networks focused in this dissertations is not static and change over time.

2.3 Datasets

2.3.1 Tweets2011 Corpus

We evaluated our proposed method of IR using the test collection for the TREC 2011 and 2012 microblog track (Tweets2011 corpus[1]). This collection consists of about 16 million tweets sampled between January 23 and February 8, 2011, for 110 search topics. Figure 2.1 presents a topic from the TREC 2011 and 2012 microblog tracks. In the figure, $\langle num \rangle$ is a topic number, $\langle title \rangle$ is a user query,

<num>	MB001
<title>	BBC World Service staff cuts
<querytime>	Tue Feb 08 12:30:27 +0000 2011

Figure 2.1: Example topic from the TREC microblog track.

[1]http://trec.nist.gov/data/tweets/

Table 2.1: Summary of TREC collections and topics used for evaluation.

Name	Type	#Topics	Topic Numbers
TREC 2011	*allrel*	49	1-49
	highrel	33	1, 10-30, 32, 36-38, 40-42, 44-46, 49
TREC 2012	*allrel*	59	51-75, 77-110
	highrel	56	51, 52, 54-68, 70-75, 77-104, 106-110

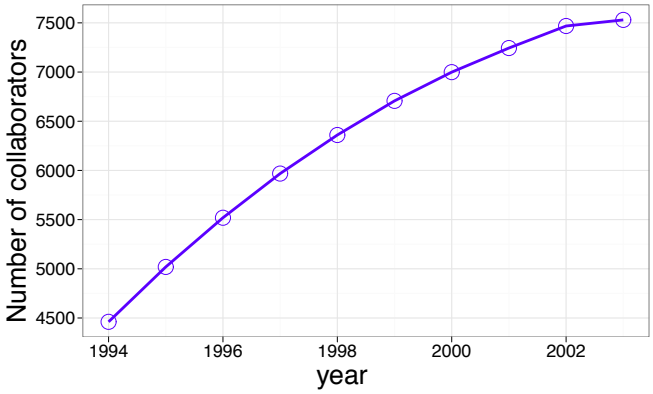

Figure 2.2: The number of collaborations in a co-authorship network of arXiv (hep-th) from 1994 to 2003.

and $\langle querytime \rangle$ is the query-time when the query was issued. In our experiments, we use $\langle title \rangle$ as a test query which is the official query used in the TREC 2011 and 2012 microblog track.

To evaluate any IR system, relevance judgment is applied to the whole tweet set of each topic. The relevance levels are categorized into irrelevant (labeled 0), minimally relevant (labeled 1), and highly relevant (labeled 2). We separately evaluate our method with respect to *allrel* and *highrel* query sets: *allrel* has both minimally relevant and highly relevant tweets as relevant documents and *highrel* has only highly relevant tweets. Table 2.1 summarizes topic numbers that we used in our experiments.

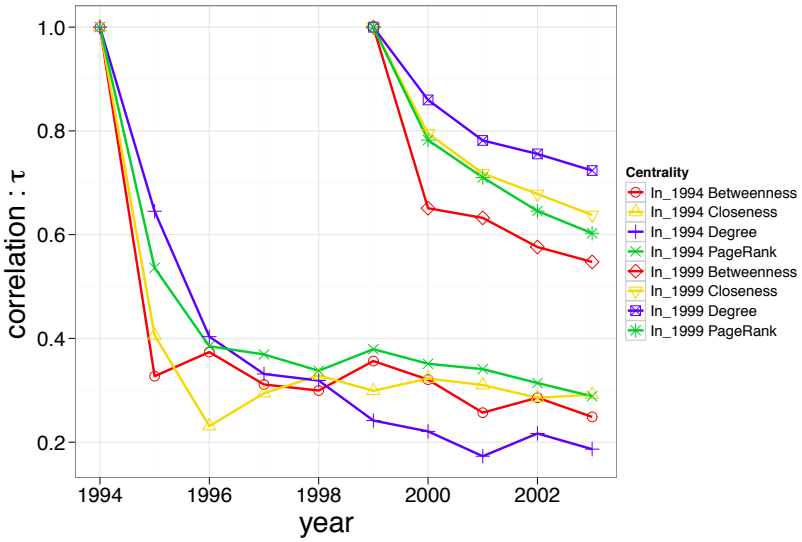

Figure 2.3: Author rank (top=100, 2000) with every network centrality by year.

2.3.2 Hep-Th

We rank objects according to the network centrality and show that such objects' ranking changes over time. We assume that we need to predict object ranking in a dynamically changing network.

To validate the change of object ranking, we use a co-authorship network from the arXiv(hep-th) [75] citation dataset. This dataset includes 8,392 authors and 87,794 co-authorships from 1900 to 2003. The number of co-authorship is 20,387 without duplicative co-authorships. Figure 2.2 shows the number of co-authorships from 1994 to 2003. From this figure, we found that the number of collaborations in the network increases over time.

Then, we tracked the ranking of all 2,950 authors appearing in 1994 to examine if object ranking in the following years would change. For example, we compare the ranking in 1994 to the ranking from 1995 to 2003 with Kendall's τ [35]. The rank correlation $\tau = 1$ denotes that two rankings are entirely the same and $\tau = 0$ denotes that rankings are completely different. Figure 2.3 shows the change of object ranking by the network centrality, i.e., Betweenness, Closeness, Degree, and PageRank. The illustrated rankings in Figure 2.3 are restricted to top 100 and 2000 objects. For all the centralities, the correlation τ decreases over time. The result suggests that the ranking of objects at a given point will change in the future. Thus, we need to predict importance of object for finding promising objects in a network.

2.4 Summary

In this chapter, we summarized the background and the previous work related to this dissertation in Section 2.1. Then, in Section 2.2, we introduce language modeling framework for IR, previous time-aware IR methods, and object ranking methods by the network centrality. Finally, Section 2.3 described the datasets for experiments in this dissertation. In the following chapters of this dissertation, we will evaluate the empirical results of our work using these datasets.

Chapter 3

Word-Based Temporal Relevance Feedback

3.1 Introduction

As mentioned in Chapter 1, not all of the temporal variations of a given topic in social media are the same. Moreover, existing IR methods cannot effectively combine recency and the temporal variation of a topic in accordance with the type of its temporal variation. To overcome the limitations of existing methods, we build time-based query expansion (QE) methods that can handle recency and ones that can handle temporal variation. Moreover, we combine these QE methods to compensate for the limitations of the individual methods and improve retrieval performance by automatically detecting a topic's temporal variation. In addition, our method enables visual analysis of when a topic and topic-related terms are actively mentioned. We used the Tweets2011 corpus[1], which consists of more than 16 million tweets over a period of two weeks to verify the effectiveness of our method.

The remainder of this chapter is organized as follows. First, in Section 3.2, we show the previous time-based microblog search method. Then, in Section 3.3, we describe the proposed method combining two types of temporal properties for a word-based query expansion. In Section 3.4 we empirically evaluate the performance of the proposed temporal word-based query expansion. We conclude the chapter in Section 3.5.

[1] http://trec.nist.gov/data/tweets/

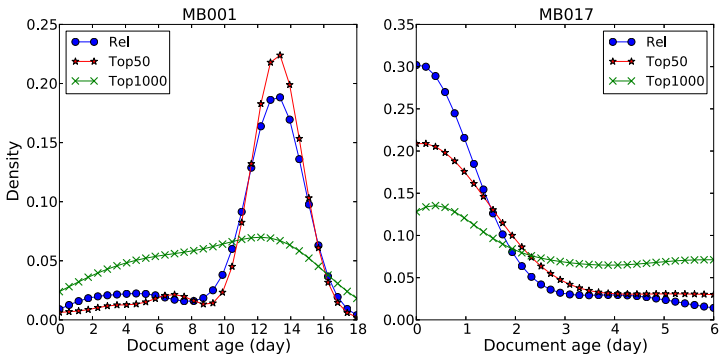

Figure 3.1: Two kernel density estimates corresponding to topics MB001 and MB017. The blue line (*Rel*) is the estimate for relevant documents. The red line (*Top50*) and green line (*Top1000*) are the estimates for the top 50 and 1000 retrieved documents, respectively.

3.2 Previous time-based microblog search method

Microblog users often search for documents regarding a recent topic concerning an event that happened recently. Documents relevant to some recent topics tend to be issued at around the query time (e.g., MB017 and MB026 in Figure 1.1). Taking advantage of this characteristic, Efron [21] incorporated temporal properties, such as recency and the smoothed temporal variation of a topic, into microblog search. His method used a temporal profile [29] represented as a timeline for a set of documents returned by a search engine and assumed that the density of a relevant document's temporal profile (relevant profile) has a smaller Kullback-Leibler (KL) divergence from the temporal profile for a seed query (query profile) than the non-relevant document's profile (irrelevant profile). Efron's idea is exemplified in Figure 3.1 which shows the kernel density estimates based on three temporal profiles (*Rel*, *Top50*, and *Top1000*) using different tweet sets: relevant tweets and top 50 and 1000 tweets retrieved by Indri search engine with default settings. Here, *Rel*, *Top50*, and *Top1000* are regarded as the relevant profile, query profile, and irrelevant profile, respectively, since the evaluation values of precision at 50 with MB001 and MB017 (0.74 and 0.36, respectively) are significantly higher than the values of precision at 1000 (0.061 and 0.064); thus, we confirmed that the shape of the relevant profile *Rel* is more similar to the query profile *Top50* than to the irrelevant profile *Top1000*. By leveraging this temporal property, Efron re-ranked documents according to the following score:

$$s(D,Q) = \log P(Q|D) + \phi(T_Q, T_D), \tag{3.1}$$

where $\phi(T_Q, T_D) = \log(\frac{m_{T_Q}}{m_{T_D}})$ and m_{T_Q} represents the sample mean of time-stamps (average document age) extracted from the documents retrieved by query Q, and m_{T_D} is the sample mean of the time-stamps extracted from the documents retrieved by a pseudo-query D, which is a document retrieved by query Q. The small sample mean m_{T_D} promotes new documents and penalizes old ones. The penalty is tempered if query Q shows weak preference for recent documents.

Efron's model, however, cannot identify terms related to a query and cannot handle multimodal temporal variations (e.g., those for MB021 and MB026 in Figure 1.1) since it assumes that time-stamps are generated from a Gaussian distribution. Our model for handling any temporal variations and discovering terms temporally related to a topic for QE is explained in Section 3.3. It ingeniously combines two types of time-aware QE methods according to the temporal variation of a given topic.

3.3 Proposed Method

In this section, we describe how to leverage temporal properties in order to refine a seed query. We present several QE methods utilizing various temporal properties (as described in Figure 1.1). The following outlines our QE method.

1. Extract time-stamps from a set of tweets returned by a search engine with a seed query and build a temporal profile (query profile).
2. Choose candidate terms for QE in the top M tweets.
3. Re-retrieve tweets using *both* the seed query and the candidate term as an expanded query and build a temporal profile (expanded query profile).
4. Use the temporal profiles for two types of QE methods: recency-based and temporal-variation-based methods.
5. Combine the scores of the two types of temporal QE methods according to the temporal variation of the query profile.
6. Re-retrieve tweets using an expanded query with K candidate terms ordered by the integrated score and remove retweets[2] from the tweets.

[2]Tweets re-posted by another user to share information with other users

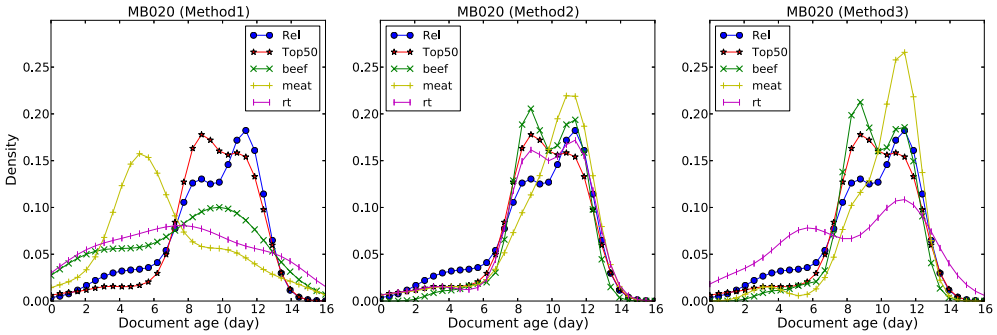

Figure 3.2: Three types of kernel density estimates obtained using topic MB020 (Taco Bell filling lawsuit). Green, yellow, and purple lines show the temporal profiles for *beef*, *meat*, and *rt*, respectively. *Top50* and *Rel* are temporal profiles created from the top 50 documents and relevant documents for the topic.

3.3.1 Temporal Profile for Query Expansion

In this section, we describe a QE method that adds topic-related terms to a seed query. Figure 3.1 shows that the query profile (*Top50*) can be regarded as an approximation of the relevant profile (*Rel*). Our assumption is that we can identify terms related to a given topic by comparing the query profile with the expanded query profile. To confirm this idea, we tried out three types of retrieval methods as follows: **Method1** retrieves documents with only one candidate term as a query. **Method2** retrieves documents that contain at least one seed query term or a candidate term. **Method3** retrieves documents that contain *both* at least one seed query term and a candidate term. We use the query likelihood model [64] with Dirichlet smoothing [83] (we set smoothing parameter $\mu' = 2500$) implemented by the Indri search engine to retrieve documents for building temporal profiles. All queries and tweets are stemmed using the Krovetz stemmer without stop-word removal and are case-sensitive. For all methods, the temporal profile for non-related terms must not be similar to the relevant profile in order to distinguish related terms from non-related terms. To determine an appropriate method that can find related terms, we used three temporal profiles about the topic "*Taco Bell filling lawsuit*" (MB020). The temporal profiles of three terms: *beef*, *meat*, and *rt* are also described. Two terms *beef* and *meat* are related to the topic since the news about the lawsuit of Taco Bell's augmented beef, *Taco Meat Filling*, was reported in late January 2011[3]. On the other hand, *rt* is a general term denoting a retweet, so it is not related to any particular topic. The results of each method are indicated in Figure 3.2. The left plot (**Method1**) shows that the temporal pro-

[3]http://gizmodo.com/5742413/

file for *beef* is incorrectly similar to the profile for *rt* than the profiles for *Rel* and *Top50* are. Furthermore, the temporal profile for *meat* deviates from the relevant profile since *meat* matches irrelevant documents; thus, **Method1** tends to retrieve tweets describing other topics and makes it difficult to detect topic-related terms correctly. The center plot (**Method2**) shows that all temporal profiles are similar to the profile *Top50* and the profile *Rel* owing to the number of seed query terms. If the number of seed query terms is large, the weight of the query likelihood of seed query terms in the expanded query become higher than a candidate term since the query likelihood model [64, 83] gives a higher ranking to documents that contain the query terms. As a result, **Method2** unfortunately tends to retrieve tweets include more query terms and makes similar temporal profiles, so this method has poor ability to identify topic-related terms for some queries. The right plot (**Method3**) shows that the temporal profile created from the combination of a seed query and a related term (e.g., *beef* and *meat*) is similar to that of the relevant profile (*Rel*). In contrast, the temporal profile corresponding to a general term (*rt*) deviates from that of relevant documents since expanded queries "*filling lawsuit rt*" and "*Taco Bell rt*" tend to retrieve tweets mentioning various topics about *filling lawsuit* and *Taco Bell* compared with an expanded query "*Taco Bell beef*" including both query terms and the topic-related term and can search tweets about the intended topic. From these observations, we conclude that **Method3** is effective at building a temporal profile for selecting appropriate candidate terms for QE; at least, for this topic (although **Method3** works better than other methods for other many topics, this cannot be discussed here owing to a lack of space). Hereinafter, we use **Method3** for making the expanded query profile.

Temporal modeling

To model the temporal properties of a candidate term combined with a seed query, we borrow Jones and Diaz's idea [29]. At first, the distribution in a particular day t is defined as $P(t|Q)$, where Q is a query. This probability is defined as

$$P'(t|Q) = \sum_{D \in R} P(t|D) \frac{P(Q|D)}{\sum_{D' \in R} P(Q|D')},$$

where R is the set of top M documents returned by a search engine for Q, D is a document, and $P(t|D) = 1$ if the dates of t and D are the same; otherwise, $P(t|D) = 0$. Here, $P(Q|D)$ is the relevance score of a document D for Q.

To handle possible irregularity in the collection distribution over time, background smoothing is applied as follows:

$$P(t|Q) = \lambda P'(t|Q) + (1 - \lambda) P(t|C),$$

where the temporal model of this collection C (the collection temporal model) is defined as $P(t|C) = \frac{1}{|C|}\sum_{D\in C} P(t|D)$; here, C is the set of all documents in a corpus. We set λ to 0.9 following previous work [29] and use this $P(t|Q)$ as the query temporal model. Although the existing method applies smoothing across adjacent days for the query temporal model, we do not do so in our microblog search settings since the daily frequency of a term is important for a microblog.

Temporal variation

By measuring the difference between the query profile and the expanded query profile (temporal profile created from an expanded query), we devised a new QE method (TVQE) for selecting temporally related terms. This model is based on the insight derived from Figure 3.2 (right plot), where the temporal profile created from the combination of a seed query and a related term is similar to the relevant profile and conversely that the temporal profile of a non-related term is dissimilar to the relevant profile. The candidate terms are selected by the following KL-divergence between two temporal models.

$$S_{TVQE}(w,Q) = -D_{KL}(P(t|w\cap^{+}Q), P(t|Q)) = -\sum_{t=1}^{T} P(t|w\cap^{+}Q)\log\frac{P(t|w\cap^{+}Q)}{P(t|Q)},$$

where $w\cap^{+}Q$ is the expanded query (produced by **Method3** in Section 3.3.1) that includes *both* at least one seed query term and a candidate term. We assume that a term with low KL-divergence for a seed query that has the ability to retrieve relevant documents as effectively as a seed query. This is because low KL-divergence indicates that a candidate term has been used along with at least one seed query term over time. Moreover, our model can capture daily document frequency, so it is applicable to any temporal variations. However, it unfortunately ignores the recency factor.

Temporal recency

To incorporate recency into a QE method, we also use another QE method (TRQE), which is a modification of Efron's model (see Equation (3.1)) as follows:

$$S_{TRQE}(w,Q) = \phi(T_Q, T_{Q'}) = \log\Big(\frac{m_{T_Q}}{m_{T_{Q'}}}\Big), \tag{3.2}$$

where $m_{TQ'}$ is the sample mean of the time-stamps (average document age) obtained from the top L documents retrieved by a search engine with a query that includes a term w and at least on seed query term. This model can suggest the candidate term related to a given query, which favors more recent documents than

a seed query; on the other hand, original Efron's model cannot discover related terms.

3.3.2 Combined Query Expansion

As described in the previous sections, all the methods have strengths and weaknesses. TRQE can incorporate temporal properties, especially recency, into models to easily detect recent documents relevant to a topic (e.g., MB017 and MB026 in Figure 1.1), but these models only partially consider when a topic is actively mentioned (e.g., MB001 and MB021 in Figure 1.1). In contrast, TVQE can manage such temporal variation by introducing temporal profiles and find the expanded query that has similar temporal profile to a seed query. However, it ignores recency.

Temporal variation + recency

To solve this problem, we combine two types of temporal properties—temporal variation and recency—by leveraging the characteristic of a query profile. As we have shown in Figure 3.1, the query profile approximately represents the relevance profile (real temporal variation of a topic). In modeling the topic temporal variation, we assume that all time-stamps of documents are generated from Gaussian distributions. To find a topic's temporal variation type, we estimate the probability ζ of a random variable X (time-stamp of tweet) falling in the interval $(-\infty, \gamma]$ using a cumulative density function as follows:

$$\zeta = P(X \leq \gamma) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\gamma} \exp\left\{\frac{(x-\mu)^2}{2\sigma^2}\right\} dx, \tag{3.3}$$

where μ denotes the mean, and σ^2 is the variance of the Gaussian distribution. We estimate the parameters μ and σ^2 by maximum-likelihood estimation (MLE); MLE can select the best model and parameters to explain the observed data (document time-stamps in our case), so we can approximately model the topic's temporal variation. Note that the probability ζ means how many tweets were generated by users until γ days after the topic's query time. If the query profile of a given query has many documents generated at around its query time, the probability of the query is high; on the other hand, the probability is low if those document time-stamps are far from the query time. For example, the probabilities of topics MB001 and MB017 (shown in Figure 3.1) until $\gamma = 6$ days after of the query time are 0.024 and 0.945, respectively, when we use the parameters $\mu = \mu_{MLE}$, $\sigma^2 = \sigma^2_{MLE}$ estimated by MLE using the time-stamps of tweets retrieved by each seed query.

By using the probability ζ, our combined method (TVRQE) automatically weights two types of QE methods, TVQE and TRQE, as follows:

$$S_{TVRQE}(w,Q) = (1-\zeta) \cdot S'_{TVQE}(w,Q) + \zeta \cdot S'_{TRQE}(w,Q)$$

where $S'_{TVQE}(w,Q)$ and $S'_{TRQE}(w,Q)$ are the standard scores of $S_{TVQE}(w,Q)$ and $S_{TRQE}(w,Q)$, respectively. The weight of $S'_{TVQE}(w,Q)$ is high if the query profile is built far from the query time; on the other hand, the weight of $S'_{TRQE}(w,Q)$ is high if the query profile of a given topic is built at around the query time.

3.4 Evaluation

3.4.1 Experimental Setup

In this section, we explain the test collection in the TREC 2011 microblog track (Tweets2011 corpus) used to evaluate our method. In addition, relevance judgment is applied to the whole tweet set of each topic. The relevance levels are categorized into irrelevant (labeled 0), relevant (labeled 1), and highly relevant (labeled 2). In all our experiments, we considered tweets labeled 1 and 2 as relevant and others as irrelevant.

Tweet collection

We indexed tweets posted before the specific time associated with each topic by the Indri search engine with the setting described in Section 3.3.1. This index was created to simulate a realistic real-time search setting, where no future information is available when a query is issued. We built an index for each query. In our experiments, we used the titles of TREC topics numbered 1–50[4] as test queries, which are the official queries in the TREC 2011 Microblog track. For retrieving documents, we used a basic query likelihood model with Dirichlet smoothing ($\mu' = 2500$) as the likelihood model (LM) and all retrievals used this LM. Note that retweets were regarded as irrelevant for evaluation in the TREC 2011 Microblog track; however, we used retweets except a final ranking of tweets since some retweets contain relevant tweets and thus are important clues for identifying appropriate expansion terms. In the final ranking, retweets were removed and all non-English retrieved tweets were filtered out by using a language detector with infinity-gram, called ldig[5].

[4]The topic numbered MB050 has no relevant tweets, so we did not use it for our experiments.

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

Table 3.1: Retrieval performance of the QE method (we set $K = 10$, $L = 30$, $M = 30$, $\gamma = 5$). Metzler [51] is the best performance for the realtime adhoc task in the TREC 2011 Microblog track. Liang [43] is a state-of-the-art query modeling approach post TREC 2011.

Method	LM	RF [40]	RQE [50]	TVQE	TRQE	TVRQE	Metzler	Liang
P@30	0.4218	0.4503	0.4619	0.4605	0.4707	**0.4830** †	0.4551	0.4490
MAP	0.2484	0.2585	0.2690	0.2679	0.2656	**0.2741**	—	0.2552

For QE, we re-retrieved tweets with an expanded query consisting of a seed query and K candidate terms extracted from the top M tweets retrieved by the seed query. We selected the candidate terms in the top 30 tweets ($M = 30$) retrieved by the seed query. Then, we selected candidate terms among tweets after removing the uniform resource locators (URLs), users names starting with '@', and special characters (!, @, #, ', ", etc.). All query terms, candidate terms, and tweets were decapitalized. The candidate terms did not include any stop-words prepared in Indri. For TVQE and TRQE, we used the temporal profile consisting of the top 30 retrieved tweets ($L = 30$). Note that we removed candidate terms that did not appear more than five times along with a query term. All QE methods selected 10 terms ($K = 10$) among candidate terms in descending order of score estimated by each QE method. The selected terms did not contain any seed query terms. We used the combination of a seed query and the selected terms as an expanded query; they were weighted by the Indri query language [73] with $6 : 4$ for all retrievals using QE since most QE methods using this setting performed well in the preliminary experiments. The sensitivity of some parameters K, M, and L for QE is discussed in the next section.

Evaluation measure

The goal of our system is to return a ranked list of tweets by using the expanded query produced by the QE method. The evaluation measures that we used include precision at rank 30 (P@30) and mean average precision (MAP). P@30 was the official Microblog track metric in 2011 [61]. Note that we used only the top 30 tweets retrieved by each method. To test for statistical significance, we used a paired t-test. The best performing run is indicated in bold and significant improvements are indicated with † and ‡ for $p < 0.05$ against a pseudo-relevance feedback method (RF) [40], which are an Indri's implementation and a recency-based QE method (RQE) for microblog search [50] with the past work's parameters, respectively. RF is a topical QE baseline and RQE is a temporal QE baseline.

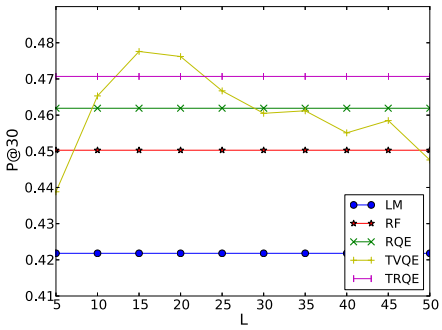

Figure 3.3: Length of the temporal profile.

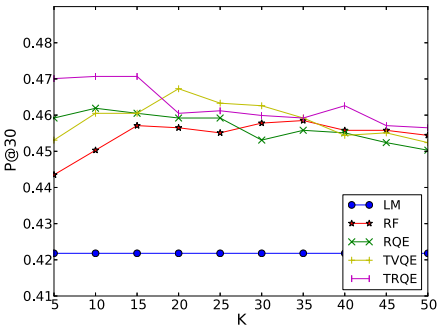

Figure 3.4: Number of candidate terms for QE.

3.4.2 Experimental Results

Overall results

In this section, we empirically evaluate our approach using 49 test topics and their relevant tweets used in the TREC 2011 Microblog track. Table 3.1 shows the results of the initial retrieval (LM), two baselines (RF, RQE), our methods (TVQE, TRQE, and TVRQE), the TREC 2011 Microblog track official result (based on learning to rank), and other results (based on temporal query modeling) reported at the post-TREC conference. Our temporal-based methods (TVQE and TRQE) resulted in improvements of 9% and 11%, respectively, in P@30 over LM. This supports the idea that using temporally related terms for QE is effective for finding documents relevant to a topic. Moreover, the combination of two types of temporal QE methods (TVRQE) outperformed strong baseline QE methods RF and RQE and others in P@30 and in mean average precision. This indicates that combining recency and temporal variation into a QE method is an effective way to improve retrieval performance of a microblog search.

Parameter sensitivity

The degrees of relationship among the parameters (L, K, M, and γ) of each QE method are shown in Figure 3.3, 3.4, 3.5, and 3.6. The x-axis shows each parameter. The y-axis shows the values in P@30. Figure 3.3 shows P@30 values for TVQE and TRQE over all topics (MB001–MB049) and for $M = 30$ and $K = 10$ across several L values. The P@30 value of TVQE was affected by the length of the query profile. TVQE with around $L = 15$ and 20 performed well because most of the relevant tweets were ranked at the top and $L = 5$ and 10 were too

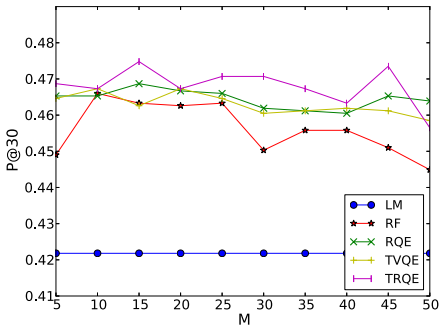

Figure 3.5: Number of tweets at the top.

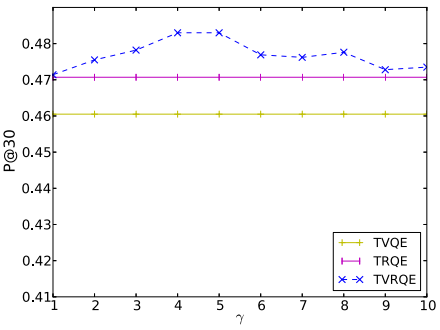

Figure 3.6: TVRQE parameter γ.

short to represent temporal variation. Interestingly, the P@30 value of TRQE was robust with respect to the query profile length owing to its definition using only the mean of the time-stamps of the query profile. TVQE and TRQE outperformed RF and RQE for several parameters. Figure 3.4 shows P@30 values of all QE methods for $M = 30$ and $L = 30$ across several K values. The results show that TRQE is a remarkable QE method because it had high P@30 values with a small K. Figure 3.5 shows P@30 values for all QE methods for $L = 30$ and $K = 10$ across several M values. The small M means a small number of candidate terms, so the retrieval performances of QE methods were almost the same for $M = 10$. RQE and TVQE were insensitive to parameter M. Figure 3.6 shows the relatedness among the P@30 values of TVRQE for $M = 30$, $L = 30$, and $K = 10$ across several γ values shown in Equation (3.3), which determine the weights of TVQE and TRQE in TVRQE. The results show that TVRQE outperformed TVQE and TRQE for all values of parameter γ.

Temporal analysis

To analyze the effectiveness of our methods (TVQE, TRQE, and TVRQE) in terms of temporal aspects, we present three types of temporal profiles: query profile, expanded query profile, and relevant profile. Figure 3.7 shows kernel density estimates of the temporal profiles for four topics: "*2022 FIFA soccer*" (MB002), "*NSA*" (MB006), "*Egyptian protesters attack museum*" (MB010), and "*release of The Rite*" (MB014). For three of these topics (MB002, MB010, and MB014), TVQE improved retrieval performance in P@30 (from 0.3000 to 0.5333, from 0.4667 to 0.8000, and from 0.4667 to 0.6000, respectively) versus the initial retrieval likelihood model; on the other hand, TVQE decreased the P@30 value for MB006 (from 0.3333 to 0.2667). Interestingly, we found that the expanded query

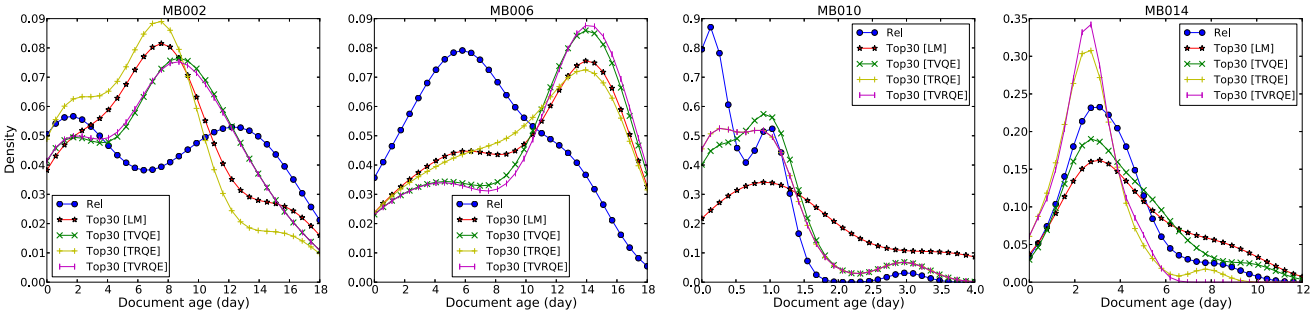

Figure 3.7: Kernel density estimates corresponding to four topics: MB002, MB006, MB010, and MB014. The curves, *Rel*, *Top30 [LM]*, *Top30 [TVQE]*, *Top30 [TRQE]*, and *Top30 [TVRQE]*, are estimates for relevant documents; the top 30 documents retrieved by using a seed query and the top 30 re-retrieved documents retrieved by using an expanded query with TVQE, TRQE, and TVRQE, respectively.

profiles (*Top30 [TVQE]*) for the former topics were similar to their relevant profiles (*Rel*); in contrast, *Top30 [TVQE]* for the latter topic was further away from *Rel*. That is because TVQE highly depends on the temporal profile obtained by a seed query, so it could estimate an expanded query profiles more similar to the relevant profile than the query profile for MB002, MB010, and MB014, which have small KL divergence between a query profile and a relevant profile; on the other hand, TRQE could not improve the P@30 value more than TVQE in MB002 owing to the limitation imposed by its inability to model multi-modal temporal variation. However, TRQE, which favors terms in recent documents, could outperform TVQE in MB014 since the time-stamps of the relevant documents for a topic were temporally closer to its query time. We found that TVRQE could combine two temporal profiles derived from TVQE and TRQE into one (*Top30 [TVRQE]*) according to the shape of the initial query profile (*Top30 [LM]*).

Examples of the expanded query

Table 3.2 lists the top 10 candidate terms suggested by three QE methods (TVQE, TRQE, and TVRQE) for four test topics: MB002, MB006, MB010, and MB014. The candidate terms were ordered by the score calculated by each QE method. We noticed that incorporating only one temporal property into a QE model was insufficient. The recency-based method TRQE could not find related terms (e.g., *qatar*, *world*, and *cup* in MB002[6]) that temporal-variation-based method TVQE

[6] 2022 FIFA World Cup will be held in Qatar.

Table 3.2: Top 10 candidate terms suggested by each QE method.

MB002			MB006			MB010			MB014		
TVQE	TRQE	TVRQE	TVQE	TRQE	TVRQE	TVQE	TRQE	TVRQE	TVQE	TRQE	TVRQE
fifa	neck	fifa	com	ng	com	secure	looters	looters	heard	topped	box
cups	governing	plans	news	rt	news	jan25	stealing	stealing	films	box	hopkins
cup	body	stage	security	google	security	jazeera	cabinet	cabinet	film	thriller	anthony
world	plans	soccer	sa	watch	google	al	human	human	2011	office	made
qatar	stadiums	qatar	nsa	nsa	nsa	shield	museums	museums	tell	made	office
2022	torres	2022	google	com	sa	shields	museum	museum	ap	zone	top
soccer	sunderland	world	former	relationships	former	looted	tanks	tanks	good	anthony	horror
best	ban	cups	apple	relationship	apple	looting	looted	looted	take	hopkins	topped
2010	stage	cup	global	news	rt	museums	looting	looting	takes	skype	thriller
winter	sepp	sepp	rt	sa	global	museum	cheered	cheered	great	ss3	heard

ranked at the top since TRQE could not precisely estimate the relevant temporal profile having a multimodal shape. The definition of TRQE in Equation (3.2) assumed that document time-stamps are generated from a unimodal distribution. However, TRQE was effective for the queries whose relevant documents existed at around the query time. For MB010, TRQE suggested topic-related terms (e.g., *looters* and *stealing* in MB010[7], *relationship* in MB006[8], and *anthony* and *thriller* in MB014[9]) that improved the P@30 value while TVQE could not. TVRQE could suggest the topic-related terms predicted by both TRQE and TVQE at the top.

3.5 Summary

Microblog users search for posts about a recent topic to understand what is happening around the world. As a consequence, information at the time that a topic is actively mentioned is an important clue for finding topic-related terms and relevant documents. In this chapter, we described three QE methods: two individual methods based on temporal variation and recency (TVQE and TRQE) and their combination (TVRQE). To overcome the limitations of the individual methods, TVRQE combines two types of temporal QE methods according to the topic's temporal variation. Our experimental results using the Tweets2011 corpus indicate that temporal properties are important features for discovering terms related to a topic and that TVRQE, which combines two time-sensitive methods, efficiently improves the retrieval performance in both P@30 and the mean average precision.

[7]The looters broke into Cairo's famed Egyptian Museum, ripping the heads off two mummies and damaging about 10 small artifacts in late January 2011.

[8]Google-National Security Agency (NSA) relationship was mentioned in early February.

[9]The movie staring Anthony Hopkins was released on January 28, 2011.

Nevertheless, the proposed temporal relevance feedback method mainly uses word frequency and do not use multi-term concepts (e.g. terms or phrases) even though such concepts can discriminate between relevant and non-relevant documents better. In the next chapter, to address this problem we introduce a concept-based temporal relevance model that uses the temporal variation of concepts on microblogs.

Chapter 4

Concept-Based Temporal Relevance Feedback

4.1 Introduction

Time plays an important role in retrieving relevant and informative microblogs because of the real-time feature of microblog documents [20, 22, 44, 63]. Particularly, query expansion methods based on relevance feedback incorporating the temporal property of words into their models have been demonstrated as effective for improving microblog search performance [15, 50, 52, 57, 58]. These time-based query expansion methods mainly use word frequency in pseudo-relevant documents as lexical information and temporal variations of word frequency as temporal information.

However, such word-based pseudo-relevance feedback (PRF) methods result in limited retrieval effectiveness for retrieving highly relevant documents. The fundamental reason is that words have semantic ambiguity. Furthermore, word frequency often fails to indicate the exact time-ranges in which crowds of people are interested [57].

To overcome the shortcomings of word-based IR, several researchers have recently proposed unsupervised or supervised concept importance weighting methods [5, 6, 7, 8, 9, 39, 41, 54, 55] because concepts (e.g., terms or phrases) generally have more discriminative power than words. However, the existing concept-based IR models do not consider time, which is an important factor for microblog search, because these methods are mainly used for Web searches, which require almost no temporal information. Therefore, the open question we are tackling is the weighting of concepts effectively using temporal information.

Table 4.1: Example of expanded words and concepts for a topic *"White House spokesman replaced"* from a word-based PRF (wRM) and a concept-based temporal one (cTRM).

wRM	cTRM (Lexical)	cTRM (Temporal)
jay	jay	carney
carney	carney	jay
qantas	qantas	press secretary
new	new spokesman	jay carney
obama	new	biden spokesman

To address this question, we propose a novel concept weighting scheme based on the temporal relevance model for query expansion. The proposed model extends a state-of-the-art concept weighting approach, called Latent Concept Expansion (LCE) [55], from a temporal perspective. We call this method *time-aware latent concept expansion*, which provides a unified framework for weighting concepts using both lexical and temporal information.

To clarify differences between the existing methods and the proposed one, Table 4.1 contrasts words and concepts suggested by a standard word-based PRF method [40], wRM, a standard concept-based lexical PRF method, cTRM (Lexical) that is equal to LCE [55], and our proposed concept-based temporal PRF method using only temporal information, cTRM (Temporal), for a topic numbered MB044: *"White House spokesman replaced"* used in the TREC microblog track. This topic is related to the news that Jay Carney, who had been the chief spokesman for Vice President Joseph R. Biden Jr., took over as White House Press Secretary. Table 4.1 clarifies that the word-based PRF method wRM suggests topic-related words *jay* and *carney*. However, *jay* and *carney* often retrieve irrelevant documents because these words appear in many documents. In contrast, concept-based methods cTRM (Lexical) and cTRM (Temporal) suggest exact topic-related concepts: *new spokesman*, *press secretary*, and *jay carney*. It is particularly interesting that in this case that the PRF method using only temporal information, cTRM (Temporal), suggests more topic-related and different concepts than cTRM (Lexical). Therefore, we assume that our temporal PRF method, cTRM, integrating lexical and temporal information for selecting topic-related concepts will be more effective than a PRF method using only lexical information (e.g., LCE) as well as the standard word-based PRF method.

This chapter has two primary contributions. First, we describe a novel time-based relevance model. Our model provides a flexible framework for selecting important words and concepts associated with a specified time period. This framework is a natural extension of standard word and concept weighting schemes [40, 55]

from a temporal perspective. Second, we carry out a detailed empirical evaluation which demonstrates the state-of-the-art effectiveness of the proposed model on a standard test collection for microblog search (Tweets2011 corpus). Our evaluation shows that the proposed PRF using multi-term concepts is particularly beneficial for retrieving highly relevant documents.

The remainder of the chapter is organized as follows. Section 4.2 describes details of the proposed concept-based temporal relevance model. Experimental settings and results are presented in Section 4.3. Finally, Section 4.4 presents a summary of this chapter.

4.2 Proposed method

Microblog services often have real-time features by which many microblogs are posted by crowds of people when a notable event occurs [69]. Many reports have described the effectiveness of incorporating such real-time features into PRF methods for microblog search [15, 50, 57, 58]. Therefore, we propose a concept-based PRF method that combines lexical and temporal information of concepts.

We assume that the proposed concept-based relevant model $P(c|\mathcal{R})$ derives from both lexical and temporal information sources. Therefore, we have

$$\begin{aligned} P(c|Q) &= \sum_{D_l \in \mathcal{R}_l} \sum_{D_t \in \mathcal{R}_t} P(c, D_l, D_t|Q) \\ &= \sum_{D_l \in \mathcal{R}_l} \sum_{D_t \in \mathcal{R}_t} P(D_l|c, D_t, Q) P(c, D_t|Q), \end{aligned}$$

where D_l denotes a document from pseudo-relevant documents $\mathcal{R}_l$ and D_t denotes each time (a day in our case) in $\mathcal{R}_t$. Then, as with the work by Efron and Golovchinsky [20], we apply the simple assumption that the temporal information D_t is independent of the lexical information D_l, so that D_t is dropped from the conditional probability in Equation (4.1). Therefore, we have

$$\begin{aligned} P(c|Q) &= \sum_{D_l \in \mathcal{R}_l} P(D_l|c, Q) \sum_{D_t \in \mathcal{R}_t} P(c, D_t|Q) \\ &= \frac{1}{P(c|Q)} \sum_{D_l \in \mathcal{R}_t} P(c, D_l|Q) \sum_{D_t \in \mathcal{R}_t} P(c, D_t|Q) \\ &\propto \frac{1}{P(c|Q)} \sum_{D_l \in \mathcal{R}_t} P(D_l) P(c, Q|D_l) \sum_{D_t \in \mathcal{R}_t} P(D_t) P(c, Q|D_t) \end{aligned}$$

Then, following the notion of bag-of-concepts, we assume that query concepts $\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_m$ and concept c for query expansion are sampled identically and in-

dependently from a lexical distribution of pseudo-relevant documents, $\mathcal{R}_l$, and a time distribution of ones, $\mathcal{R}_t$ (top N retrieved documents). We have

$$P(c|Q) \propto \frac{1}{P(c|Q)} \sum_{D_l \in \mathcal{R}_l} P(D_l)P(c|D_l) \prod_j^m P(\hat{q}_j|D_l) \cdot \sum_{D_t \in \mathcal{R}_t} P(D_t)P(c|D_t) \prod_j^m P(\hat{q}_j|D_t)$$

where $P(c|D_l)$ and $P(\hat{q}|D_l)$ denote the probability of concept occurrence in document D; $P(c|D_t)$ and $P(\hat{q}|D_t)$ denote the probability of concept occurrence at time t. Then, because $P(c|Q)$ is a non-negative function, we have the score function that ranks a concept c in response to query Q as

$$S_{cTRM}(c,Q) \overset{rank}{=} \Big\{ \underbrace{\sum_{D_l \in \mathcal{R}_l} P(D_l)P(c|D_l) \prod_i^m P(\hat{q}_i|D_l)}_{\textbf{Lexical}} \cdot \underbrace{\sum_{D_t \in \mathcal{R}_t} P(D_t)P(c|D_t) \prod_i^m P(\hat{q}_i|D_t)}_{\textbf{Temporal}} \Big\}^{1/2},$$

Here $P(D_l)$ and $P(D_t)$ are uniform over all the distributions in D_l and D_t. The value of $P(c|D_t) \prod_j^{|m|} P(\hat{q}_j|D_t)$ increases when the candidate concept c and query concepts $\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_m$ were described together simultaneously in a range. Using the probabilities of concept occurrence $P(c|D_t)$ derived from document timestamps of pseudo-relevant documents $\mathcal{R}_t$, this PRF model represents real-time feature of a given topic in microblogging services. In addition, because $P(c|D_l) \prod_i^m P(\hat{q}_i|D_l)$ is equal to a factor of the standard concept-based PRF method, LCE (see Equation (2.5)), Equation (4.2) is obtained for the product of lexical concept information and a temporal one. Figure 4.1 clarifies the difference between the existing concept-based relevance modeling (LCE) and the proposed concept-based temporal relevance modeling.

To improve our estimates for $P(c|D_t)$, we also use Dirichlet smoothing as with the standard query likelihood model in Equation (2.2) because the value of query likelihood $\prod_i^m P(\hat{q}_i|D_t)$ becomes 0 when a query concept $\hat{q}_i$ does not appear over time in $\mathcal{R}_t$. We have

$$P(c|D_t) = \frac{|D_t|}{|D_t| + \mu_t} \hat{P}_{ml}(c|D_t) + \frac{\mu_t}{|D_t| + \mu_t} P(c|C), \tag{4.3}$$

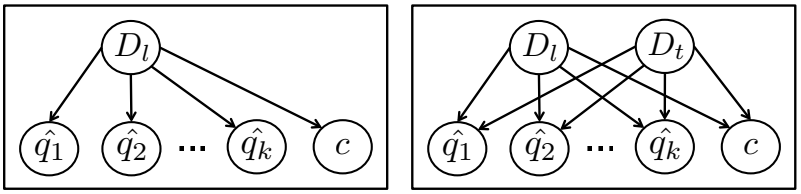

Figure 4.1: Graphical model representations of concept-based relevance modelling (left) and the proposed concept-based temporal relevance modelling (right).

where $\hat{P}_{ml}(c|D_t) = \frac{f(c;D_t)}{\sum_{c' \in \mathcal{V}_c} f(c';D_t)}$, $\mathcal{V}_c$ is the set of all concepts in the vocabulary of concepts, $f(c;D_t)$ is the frequency of concept c at time t, $|D_t|$ is the total number of concepts at time t, μ_t is a parameter for smoothing, and $P(c|C)$ is the probability of concept c occurrence in the corpus C. Finally, we rank candidate concepts in descending order of the association score $S_{cTRM}(c,Q)$ and use the top k concepts for query expansion.

4.3 Evaluation

This section describes the details of our experimental evaluation. First, in Section 4.3.1, we describe the experimental setup used for the evaluation. Then, in Section 4.3.2, we show baselines to compare our proposed method. Section 4.3.3 explains evaluation metrics and a statistical test for our evaluation. In Section 4.3.4, we compares the performance of the temporal query expansion to the performance of several standard atemporal retrieval methods. Finally, Section 4.3.5 provides additional experiments to discuss various aspects of the proposed method.

4.3.1 Experimental Setup

Evaluation data

We evaluated our proposed method using the test collection for the TREC 2011 and 2012 microblog track (Tweets2011 corpus[1]). In the figure, $\langle num \rangle$ is a topic number, $\langle title \rangle$ is a user query, and $\langle querytime \rangle$ is the query-time when the query was issued. In our experiments, we use $\langle title \rangle$ as a test query which is the official query used in the TREC 2011 and 2012 microblog track.

[1]http://trec.nist.gov/data/tweets/

Table 4.2: Summary of evaluated retrieval methods.

Method	Lexical	Temporal	Concept
wRM	✓		
cRM	✓		✓
wTRM	✓	✓	
cTRM	✓	✓	✓

To evaluate any IR system, relevance judgment is applied to the whole tweet set of each topic. The relevance levels are categorized into irrelevant (labeled 0), minimally relevant (labeled 1), and highly relevant (labeled 2). We separately evaluated our method with respect to *allrel* and *highrel* query sets: *allrel* has both minimally relevant and highly relevant tweets as relevant documents and *highrel* has only highly relevant tweets.

Microblog search settings

We indexed tweets posted before the specific time associated with each topic by the Indri search engine[2] with the following setting. All queries and tweets were stemmed using the Krovetz stemmer [36] without stop-word removal. They were case-insensitive. We built an index for each query. This index was created to simulate a realistic real-time search setting, where no future information is available when a query is issued.

To retrieve documents, we used a basic query likelihood model with Dirichlet smoothing [83] (we set smoothing parameter $\mu = 2500$ similar to Efron's work [22]) implemented by the Indri search engine [73] as the language model for IR (LM) and all PRF methods used this LM as initial search results. For temporal smoothing parameter μ_t in Equation (4.3), we set $\mu_t = 150$ when retrieving documents for *allrel* queries, and let $\mu_t = 350$ for *highrel* based on results of a pilot experiment. In addition, in stead of direct estimation of $P(c|C)$, we used $P(c|C) \approx df(c)/N$, where $df(c)$ is the document frequency of concept c and N is the total number of documents in the corpus because it can be expensive to calculate the number of documents containing a pair of query terms. Even though $df(c)/N$ is different from $P(c|C)$, we coordinate the difference with the smoothing parameter μ_t. The sensitivity of a parameter μ_t is discussed in Section 4.3.5.

We filtered out all non-English retrieved tweets using a language detector with infinity-gram, called ldig. Retweets were regarded as irrelevant for evaluation in the TREC Microblog track [61, 72]; however, we used retweets except in a final

[2]http://www.lemurproject.org/indri/

ranking of tweets because a set of retweets is a good source that might contain topic-related words for improving Twitter search performance [15]. In accordance with the track's guidelines, all tweets with http status codes of 301, 302, 403, and 404 and all retweets including the string "RT" at the beginning of the tweet were removed from the final ranking. Finally, we used the top 1000 results for evaluation.

4.3.2 IR Models

Baselines

First, we introduce the setting of the proposed PRF method. Then we describe baselines to validate the effectiveness of each component in our proposed method.

The concept-based method uses the combination of one or two words as a candidate concept. All concepts are extracted from tweets based on sequential dependence, which assumes that dependence exists between adjacent query terms [54]. Previous PRF methods also use this sequential dependence model [7, 55] because this model has consistently demonstrated state-of-the-art retrieval effectiveness in Web search. Although we use the sequential dependence model in this study, our model uses no independence structure. In addition, we used two types of concept such as #1(·) and #uw8(·), where #1(·) denotes an ordered window in which words must appear adjoiningly ordered and #uw8(·) denotes an unordered window in which all words must appear within a window of 8 terms in any order. We denote the proposed PRF method combining lexical and temporal information of concepts as cTRM.

Moreover, to assess the effectiveness of incorporating concept into the retrieval model, we also proposed a word-based temporal relevance model, wTRM, that incorporates lexical and temporal information of words into its relevance model. wTRM uses only a single word as a concept in Equation (4.2): wTRM does not consider multi-term concepts that combine more than two words. We compare this model wTRM to cTRM that uses lexical and temporal information of any concept.

To assess our proposed method cTRM, we prepared two baseline methods. The first baseline, wRM, uses a standard relevance feedback using only lexical information of words [40]. In other words, wRM uses only word information. It does not consider multiple term concepts and temporal information. Note that cTRM reduces to wRM when the number of pseudo-relevant documents from temporal perspective, $\mathcal{R}_t$, is 0 and all using concepts are single words (see Equations (2.3) and (4.2)).

Our second baseline, cRM, uses pseudo-relevance feedback with lexical information of concepts. This method is equivalent to Latent Concept Expansion (LCE) [55], except for some points. To validate the effectiveness of concept's temporal information, we use simplified LCE in Equation (2.5). This PRF model drops the penalty of the inverse collection frequency of the concept in corpus from Bendersky's LCE in Equation (2.4). Both cRM and cTRM can use any concept. However, cRM differs from cTRM in that cRM does not consider temporal information such as $\mathcal{R}_t$.

Table 4.2 summarizes the choice of concepts and pseudo-relevance information sources used by our methods and baselines. For instance, it is apparent from Table 4.2 that cRM and cTRM share the same concept types, but differ in the type of pseudo-relevant documents for concept re-weighting. Note that the PRF methods using only lexical information, wRM and cRM, are strong baselines. The PRF methods using lexical and temporal information, wTRM and cTRM, are our proposed approaches.

Query expansion

For all PRF methods, we select candidate words or concepts among the top M tweets retrieved using the original query after removing the uniform resource locators (URLs), and user names starting with '@' or special characters (!, @, #, ', ", etc.). All query terms, candidates of words and concepts, and tweets are decapitalized. The candidates of words and concepts include no stop-words prepared in the Indri search engine. Then, we select k words or concepts among candidates in descending order of the word or concept weighting score, such as $S_{wRM}(c, Q)$ or $S_{cTRM}(c, Q)$. We use the normalized score for concept weighting. For example, the weight of i-th concept is $c_i = \frac{S_{cTRM}(c_i, Q)}{\sum_j^k S_{cTRM}(c_j, Q)}$ when using cTRM. Finally, we combined the expanded concepts of PRF with their weight and the original query as an expanded query. They were weighted with 1:1. Figure 4.2 shows an example of query expansion we used. In our study, we set $\lambda_1, \lambda_2 = 0.5$.

For wTRM and cTRM, we tuned parameters: the number of pseudo-relevant documents as temporal information (i.e., N). For all methods, we also tuned their parameters: the number of pseudo-relevance feedback documents (i.e., M) and the number of expansion words (i.e., k). Values of the these parameters were optimized for best performance of Mean Average Precision (MAP) on training data because MAP is a stable measure. For example, we tuned parameters of the IR model using TREC 2012 microblog track dataset and tested it with TREC 2011 microblog dataset. In contrast, we trained the model using the TREC 2012 dataset and tested it on the TREC 2011 dataset. The sensitivity of some parameters such

```
#weight(
   λ1 #combine(bbc world service staff cuts)
   λ2 #weight(
                  c1 #1(service outlines)
                  c2 #uw8(bbc outlines)
                  c3 outlines
                  ...
                  ck #1(weds bbcworldservice)))
```

Figure 4.2: Example of query expansion of topic "*BBC World Service staff cuts*" from TREC microblog track queries.

as N in wTRM and cTRM and the number of words or concepts used for query expansion, k, is discussed in Section 4.3.5.

4.3.3 Evaluation Measure

To evaluate retrieval effectiveness, we used average precision (AP), R-Precision (Rprec), and binary preference (*bpref*). AP is the mean of the precision scores obtained after each relevant document is retrieved. Rprec is that precision after R documents have been retrieved where R is the number of relevant document for the given topic. *Bpref* considers whether relevant documents are ranked above irrelevant ones. AP and Rprec have lower error rates than Precision [12]. *Bpref* is more robust evaluation measure than AP when using incomplete relevance data [13].

To validate the retrieval effectiveness, we discuss the statistical significance of results obtained using a two-sided Fisher's randomization test [71], which is a non-parametric statistical significance test that does not assume the specific distribution. We used a Perl implementation for the randomization test[3] with 100,000 permutations and $p < 0.05$ through this chapter.

4.3.4 Experimental Results

To assess the effectiveness of our proposed methods wTRM and cTRM, we compared wTRM and cTRM using standard PRF methods: wRM and cRM.

[3]http://www.mansci.uwaterloo.ca/~msmucker/software/paired-randomization-test-v2.pl

Table 4.3: Performance comparison of the word-based PRF methods. Superscripts α, β, and γ respectively denote statistically significant improvements over LM, wRM, and wTRM. The best result per column is marked by boldface.

	allrel			*highrel*		
Method	**AP**	**Rprec**	***bpref***	**AP**	**Rprec**	***bpref***
LM	0.2936	0.3313	0.3103	0.2130	0.2286	0.1933
wRM	0.3502^{α}	0.3868^{α}	0.3594^{α}	0.2473^{α}	0.2537	0.2242
wTRM	$\mathbf{0.3726}^{\alpha\beta}$	$\mathbf{0.4089}^{\alpha}$	$\mathbf{0.3872}^{\alpha\beta}$	$\mathbf{0.2580}^{\alpha}$	$\mathbf{0.2705}^{\alpha}$	$\mathbf{0.2361}^{\alpha}$

Table 4.4: Performance comparison of the concept-based PRF methods. Superscripts α, β, and γ respectively denote statistically significant improvements over LM, cRM, and cTRM. Best result per column is marked by boldface.

	allrel			*highrel*		
Method	**AP**	**Rprec**	***bpref***	**AP**	**Rprec**	***bpref***
LM	0.2936	0.3313	0.3103	0.2130	0.2286	0.1933
cRM	0.3385^{α}	0.3725^{α}	0.3479^{α}	0.2511^{α}	0.2696^{α}	0.2356^{α}
cTRM	$\mathbf{0.3644}^{\alpha}$	$\mathbf{0.4058}^{\alpha\beta}$	$\mathbf{0.3825}^{\alpha\beta}$	$\mathbf{0.2694}^{\alpha\beta}$	$\mathbf{0.2770}^{\alpha}$	$\mathbf{0.2527}^{\alpha}$

Comparison of word-based PRF methods

Table 4.3 compares the retrieval effectiveness of the initial search (LM) and the word-based PRF method using only lexical information [40] (wRM) to the retrieval effectiveness of word-based PRF method using lexical and temporal information (wTRM), both for *allrel* and *highrel* queries. It is apparent from Table 4.3 that both wRM and wTRM markedly outperform the initial search LM on both measures across both query sets. In particular, wTRM improved search results with statistical significance in all cases. Moreover, wTRM outperformed the standard word-based relevance model wRM in terms of all evaluation measures across both query sets. The difference in AP and *bpref* for *allrel* queries was statistically significant, which suggests that incorporating temporal information through our model using single words as concepts is important for retrieving topically relevant microblogs.

Comparison of concept-based PRF methods

Table 4.4 compares the retrieval effectiveness of LM and the concept-based PRF method using only lexical information [8] (cRM) to the retrieval effectiveness of concept-based PRF method using lexical and temporal information (cTRM),

Table 4.5: Performance comparison of the standard word-based PRF method and the proposed concept-based temporal one. Superscripts α and β respectively denote statistically significant improvement over wRM, and cTRM. Best result per column is marked by boldface.

	allrel			*highrel*		
Method	**AP**	**Rprec**	***bpref***	**AP**	**Rprec**	***bpref***
wRM	0.3502	0.3868	0.3594	0.2473	0.2537	0.2242
cTRM	**0.3644**	**0.4058**	**0.3825**	**0.2694**$^{\alpha}$	**0.2770**	**0.2527**$^{\alpha}$

both for *allrel* and *highrel* queries. Table 4.4 clarifies that both cRM and cTRM markedly outperform the initial search LM on both measures across both query sets with statistical significance as with word-based approaches: wRM and wTRM. Moreover, cTRM outperformed the standard concept-based PRF method cRM in terms of all evaluation measures across both query sets. Particularly, the differences in Rprec and *bpref* for using *allrel* queries and in AP for using *highrel* queries was statistically significant. The results suggest two findings. First, latent concept expansion for pseudo-relevance feedback, which uses multi-term concepts for query expansion, is effective for microblog search. This results is consistent with previous work [51]. Second, temporal information of concepts for PRF method is an important factor for retrieving topically relevant microblog documents, so that the proposed cTRM consistently outperformed the state-of-the-art latent concept expansion method, cRM.

Comparison to the standard lexical PRF method

This section presents a comparison of cTRM with a standard word-based PRF method (wRM). Table 4.5 compares the retrieval effectiveness of the standard word-based lexical PRF method (wRM) to the retrieval effectiveness of concept-based temporal PRF method cTRM, both for *allrel* and *highrel* queries. Table 4.5 clarifies that cTRM outperformed wRM in terms of all evaluation measures across both *allrel* and *highrel* query sets. Particularly, the differences in AP and *bpref* for *highrel* queries were statistically significant, whereas there are no significant differences between wRM and wTRM for *highrel*. The results suggest the combination of using a concept instead of single word for query expansion and using a temporal information of concepts for pseudo-relevance feedback is effective to retrieve highly informative microblogs.

In conclusion, from the results in Table 4.3, 4.4, and 4.5, a microblog search system should use the concept-based temporal PRF method when searching topically and highly informative relevant documents instead of the word and concept-based

Table 4.6: Performance comparison of the existing temporal PRF methods and the proposed temporal ones. Statistically significant difference of wTRM and cTRM over the baselines are marked using α, β and γ, for EXRM [42], TBRM [32], and QDRM [58] baselines, respectively. Best result per column is marked by boldface.

	allrel			*highrel*		
Method	**AP**	**Rprec**	***bpref***	**AP**	**Rprec**	***bpref***
EXRM	0.3560	0.3846	0.3634	0.2433	0.2485	0.2202
TBRM	0.3539	0.3862	0.3607	0.2347	0.2384	0.2071
QDRM	0.3568	0.3829	0.3642	0.2522	0.2622	0.2306
wTRM	**0.3726**	**0.4089**	**0.3872**	0.2580	0.2705^{β}	0.2361
cTRM	0.3644	0.4058	0.3825	$\mathbf{0.2694}^{\alpha\beta}$	**0.2770**	$\mathbf{0.2527}^{\alpha\beta}$

lexical PRF methods.

4.3.5 Additional Experiments

In the remainder of this section, we present further analyses of the various aspects of the proposed wTRM and cTRM methods.

Comparison to existing temporal PRF methods

In Section 4.3.4, we compared the proposed temporal PRF methods (wTRM and cTRM) to lexical ones (wRM and cRM). The experimental results shows the effectiveness of temporal PRF methods comparing to lexical ones. In this section, we compare the performance of the wTRM and cTRM retrieval methods to the performance of three time-based PRF methods employing the word weighting scheme. The first method, proposed by Li and Croft [42], incorporates recency into the relevance model of the document prior. The second method, proposed by Keikha et al. [32], automatically detects this topic-related time for incorporating the temporal property into language modeling frameworks. The third method, proposed by Miyanishi [58], combines query-dependent lexical information and document-dependent temporal information of microblogs for word weighting. For comparison, we used the search results reported by Miyanishi et al. [58]. We briefly compare their performance to wTRM and cTRM because the reported results of the comparative temporal PRF methods were optimized for best performance of Precision at top 30 measure in their paper. Table 4.6 presents a comparison between our proposed methods and three existing methods. Table 4.6 shows that wTRM is the best-performing method in both measures for *allrel* queries.

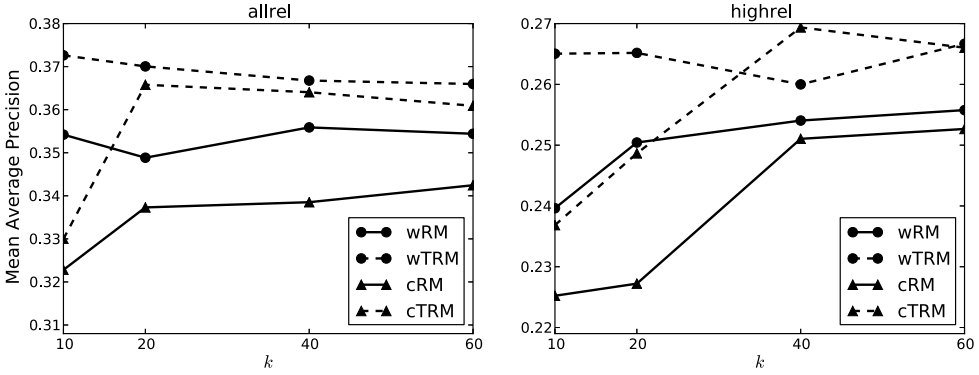

Figure 4.3: Effects of increasing the number of expansion concepts k on the retrieval effectiveness of the *allrel* and *highrel*queries. The x-axis shows parameter k. The y-axis shows the values in MAP.

Furthermore, cTRM outperformed other methods in all evaluation metrics for *highrel* queries. In particular, the difference in AP, and *bpref* for *highrel* was statistically significant. For all methods, similar queries and document processing were applied. Similar baselines were reported. Therefore, our novel PRF methods, which extended a language modeling approach from temporal perspective, are effective for microblog searches even when compared to other state-of-the-art temporal PRF methods. Moreover, Table 4.6 shows that wTRM outperformed cTRM in both measures for *allrel* queries while cTRM outperformed wTRM in both measures for *highrel* queries. Nevertheless, none of these differences was statistically significant. In summary, these results also show that concept frequencies over time are important for PRF and the concept-based PRF cTRM is an effective method to retrieve highly relevant documents.

Number of expansion concepts

In Section 4.3.4, we tuned the number of concepts k for query expansion using training data. In this section, we assess the effect of increasing the number of expansion concepts. We are particularly interested in addressing the question of whether temporal PRF methods (i.e., wTRM and cTRM) outperformed lexical ones across several k values. Figure 4.3 demonstrates that wTRM outperformed wRM, and that cTRM also outperformed cRM across several k values, which reflects that time information improves retrieval performance even when using many concepts for query expansion.

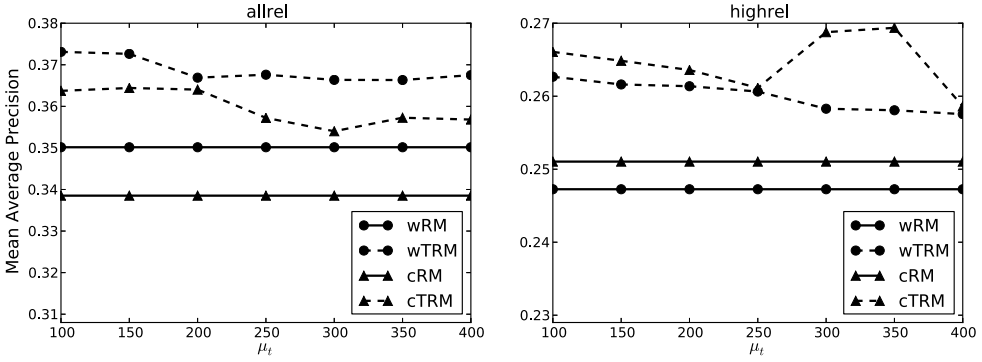

Figure 4.4: Sensitivity to a temporal smoothing parameter μ_t on the retrieval effectiveness of the *allrel* and *highrel* queries. The x-axis shows parameter k. The y-axis shows values in MAP.

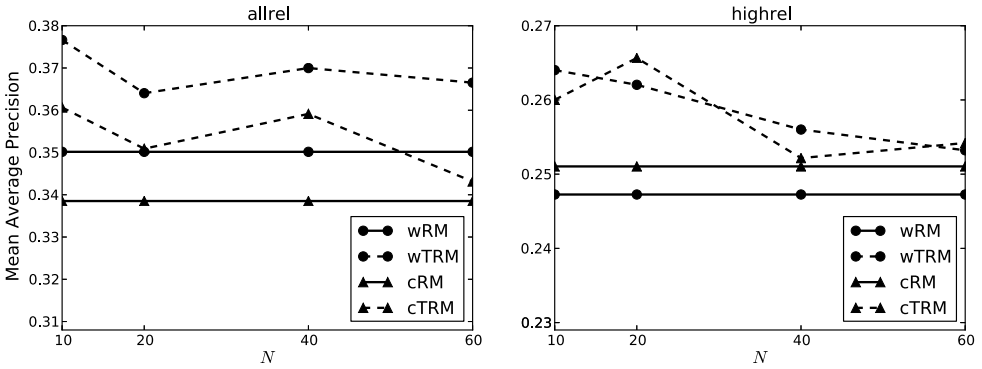

Figure 4.5: Effect of increasing the number of feedback documents for temporal information on the retrieval effectiveness of the *allrel* and *highrel* queries. The x-axis shows parameter k. The y-axis shows values in MAP.

Sensitivity to a temporal smoothing parameter

In Section 4.3.4, we let temporal smoothing parameter $\mu_t = 150$ for *allrel* and $\mu_t = 350$ for *highrel*. In this section, we assess how we should smooth language model associated with temporal information. Figure 4.4 shows that temporal methods wTRM and cTRM outperform atemporal methods wRM and cRM over *allrel* and *highrel* queries across several μ_t values. In addition, for *allrel* queries, wTRM outperformed wRM as well as cTRM across several μ_t values. However, for *highrel* queries, cTRM outperformed cRM as well as wTRM in almost all μ_t values. The MAP values of wTRM and cTRM were actually affected by the value of μ_t, which suggests that the temporal smoothing parameter μ_t requires different

tuning to achieve the best performance for *allrel* and *highrel* query sets.

Number of pseudo-relevant documents for temporal evidence

In this section, we describe our study of the effect of increasing the number of feedback documents for temporal information. The large number of feedback documents N means tracking concept's frequency over the long term. Figure 4.5 demonstrates that wTRM and cTRM respectively outperformed wRM and cRM across different feedback documents. However, their performance decreased slightly for *allrel* and substantially decreased for *highrel*, which indicates that our temporal PRF methods require few feedback documents for concept importance weighting but rather topic-related document for estimating the topically relevant time.

Expanded concepts

In this section, we present illustrative examples of the types of concepts generated using our model. Figures 4.6 and 4.7 show the top 12 expanded concepts inferred from four PRF methods (wRM, wTRM, cRM, and cTRM), respectively, for topics numbered MB109 and MB108. The expanded concepts were ordered by the score of each PRF method. Left panels in Figures 4.6 and 4.7 show the temporal variations of each topic. The x-axis shows the document age from the query-time when query was issued to document time-stamp. The y-axis shows the kernel-estimated probability density for the document age. High density indicates the period during which the topic was described actively. The solid line (Rel) shows the estimate for relevant documents. The dotted line (LM) show the estimate of top 30 retrieved documents by LM with only language filtering, which were used for temporal PRF methods.

In fact, Figures 4.6 and 4.7 clarify that estimating accurate temporal variation of a given topic using temporal PRF methods wTRM and cTRM suggests more topic-related words and concepts than wRM and cRM using only lexical information for their feedback. For example, wTRM and cTRM improved the retrieval performance in AP (0.4454 to 0.5109 and 0.4014 to 0.5843) versus wRM and cRM, respectively, because wTRM and cTRM can rank topic-related words and concepts (e.g., *film*, *documentary*, and *oscar nomination* in MB109[4]) at the top. However, wTRM and cTRM could not find topic-related words and concepts

[4] 'Gasland' is a documentary movie which has earned an Academy Award nomination for best documentary in 2011.

wRM	wTRM	cRM	cTRM
oscar	oscar	truth	oscar
industry	industry	truth gasland	industry
truth	truth	oscar	truth
nod	nod	industry	nod
fundamentally	nomination	gasland fundamentally	oscar nod
dishonest	film	dishonest	oscar nomination
moore	documentary	fundamentally dishonest	nomination
gore	moore	fundamentally	truth gasland
nomination	gore	gasland moore	film
news	news	gore	gasland moore
receives	filmmakers	more	moore gore
boos	boos	nod	gore

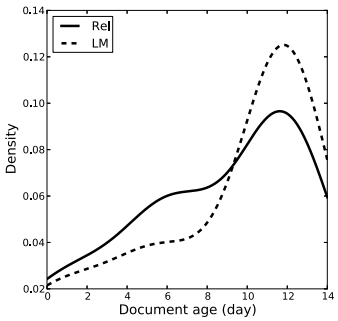

Figure 4.6: Twelve most likely one or two word concepts discovered by wRM, wTRM, cRM, and cTRM for the query "*Gasland*" (MB109), showing improved results with temporal PRF methods wTRM and cTRM. Left figure shows temporal variations of a topic numbered MB109.

wRM	wTRM	cRM	cTRM
best	best	best identity	advocate
advocate	advocate	advocate	best
cost	ring	best	advocate best
www	alleged	advocate best	best identity
hub	www	theft cost	alleged identity
restoration	hub	cost	theft ring
spears	restoration	restoration www	ring
prepaidlegal	spears	com hub	alleged
com	prepaidlegal	hub spears	com hub
colorado	com	protection restoration	hub spears
experts	colorado	www	protection restoration
scammed	experts	prepaidlegal com	spears

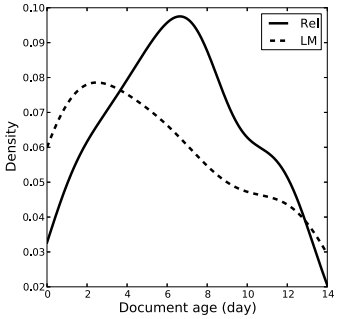

Figure 4.7: Twelve most likely one or two word concepts discovered by wRM, wTRM, cRM, and cTRM for the query "*identity theft protection*" (MB108), showing harmed results with temporal PRF methods wTRM and cTRM. Left figure shows temporal variations of a topic numbered MB108.

(e.g., *scammed*, *cost*, and *theft cost* in MB108[5]) and decreased AP values (0.3552 to 0.2185 and 0.3753 to 0.2038) versus wRM and cRM, respectively. These results suggest that estimating the relevant time for each topic is important to weight important concepts accurately.

4.4 Summary

This chapter presented a concept-based query expansion method based on a temporal pseudo-relevance feedback (PRF) model. Unlike existing retrieval models

[5]The article titled "How Much Does Identity Theft Cost?" was described by many people in Twitter around January 29, 2011.

that use only lexical information of concepts, the proposed model effectively combines lexical and temporal properties by modeling temporal variations of concepts in microblogging services. Our empirical results on the Tweets2011 corpus used in TREC 2011 and 2012 microblog track demonstrate that incorporating temporal information of concepts into the query expansion method improved retrieval performance significantly. We demonstrated that using multi-term concepts for the temporal PRF method can be useful for retrieving highly relevant documents. Furthermore, our method significantly outperformed existing temporal PRF methods.

In chapter 3 and 5, we described query expansion methods using pseudo-relevance feedback were effective for time-stamped documet search. Pseudo-relevance feedback assumes that the top ranked documents in the initial search results are relevant and that they contain topic-related words appropriate for relevance feedback. However, those assumptions do not always hold in reality because the initial search results often contain many irrelevant documents. In the next chapter, to overcome the limitation of pseudo-relevance feedback methods, we present a query expansion method based on two-stage relevance feedback approaches.

Chapter 5

Interactive Temporal Relevance Feedback

5.1 Introduction

Query expansion based on relevance feedback has been shown effective for improving microblog search performance [15, 43, 50, 52, 57]. That is due to the fact that query expansion can overcome the severe vocabulary mismatch problem of microblog search. However, classical relevance feedback, such as the Rocchio algorithm [68], requires a number of judged documents. Moreover, relevance judgment is often burdensome as it requires manually reading those documents. On the other hand, query expansion based on pseudo-relevance feedback (PRF) does not require judged documents [16, 32, 40, 42, 48, 55, 81]. The assumptions behind PRF are that the top ranked documents in the initial search results are relevant and that they include good words for query expansion. When the assumptions do not hold, PRF results in ineffective query expansion [56]—only a few of the suggested expansion words are useful and many others are either harmful or useless [14]. To overcome these problems, we propose a simple but effective query expansion method with manual selection of a single relevant document, which typically includes topic-related words. Using the selected document as query expansion words for a new query, we can re-retrieve more relevant documents and, based on the documents, estimate more accurate lexical and temporal evidence for improving the second-stage PRF described shortly. We designate this first-stage relevance feedback as *tweet selection feedback* for searching Twitter messages (i.e., tweets).

Previous works have also shown that time-based language modeling and relevance feedback approaches are effective for microblog search [15, 20, 22, 43, 50]. As

described herein, we build on these findings and propose a novel PRF method combining lexical and document-dependent temporal evidence of microblog in response to a query, which relies strongly on relevance information among the re-retrieved documents, such as a word distribution and a time-stamp distribution. We assume that the proposed PRF method further improves microblog search performance in combination with tweet selection feedback. To demonstrate the validity of our proposed approach, we carry out evaluative experiments on the datasets of the TREC 2011 and 2012 real-time ad-hoc task (i.e., Tweets2011 corpus[1]), which consist of more than 16 million tweets over a period of two weeks. The experimental results of the two-stage relevance feedback show that our tweet selection feedback reduces the adverse effects of PRF for difficult queries and is especially effective when combined with our proposed PRF.

The remainder of the chapter is organized as follows. In Section 5.2 we present the limitation of standard relevance feedback methods. Section 5.3 describes details of our proposed method, which consists of two-stage relevance feedback, tweet selection feedback and lexical-and-temporal-based relevance feedback. In Section 5.4 we demonstrate the effect of the proposed PRF methods. Finally, Section 5.5 presents a summary of this work and conclusions.

5.2 Limitation of Pseudo-Relevance Feedback

The previous time-based language models for IR and temporal relevance model based on PRF integrated into query expansion methods achieved great success for improving microblog search performance [15, 20, 22, 43, 50]. They can incorporate recency or temporal variation on microblogging platform into their model and overcome the vocabulary mismatch problem. These PRF methods assume that the proportion of relevant documents in initial search results is large, so that top ranked documents include good words for query expansion. However, that assumption becomes invalid and PRF fails if the initial search rank non-relevant documents at the top [56]. Moreover, several words suggested by PRF model are useful and many others are either harmful or useless [14]. We assume that PRF for microblog search also fails to improve search performance for some topics while enhancing the performance for other topics.

To see the performance of PRF over initial search results, we compare several PRF methods to the initial search. As the initial search, we use the language model with Dirichlet smoothing of Indri search engine[2]. We refer to this method as LM.

[1] http://trec.nist.gov/data/tweets/

[2] http://www.lemurproject.org/indri/

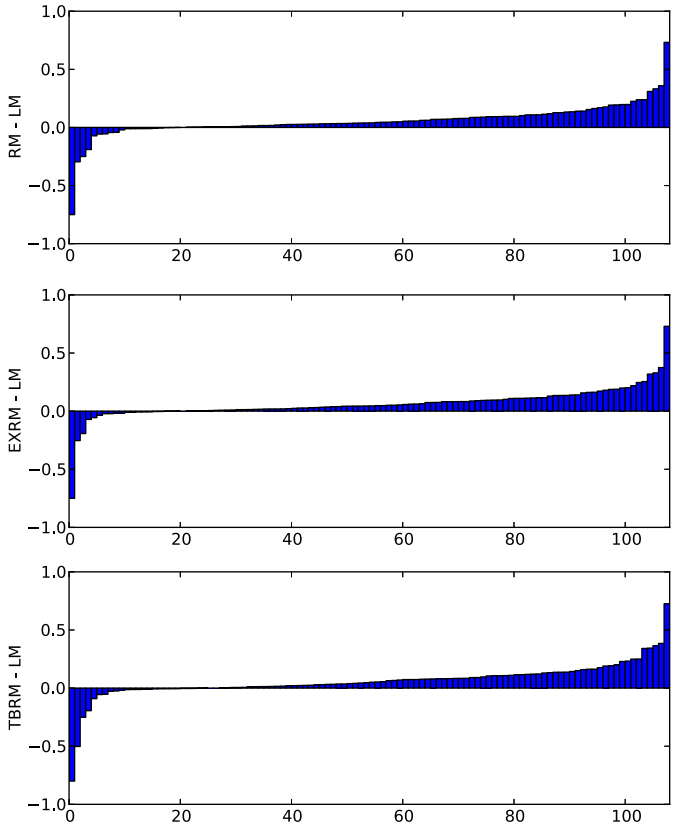

Figure 5.1: Improvements by existing relevance feedback methods over the initial search. Each bar shows the difference in average precision comparing LM to RM (top), EXRM (middle), and TBRM (bottom).

Unless otherwise specified, all retrievals are implemented on top of LM. We prepare three baseline PRF methods: the standard relevance model [40] (see Equation (2.3)), exponential recency-based relevance model [42] (see Equation (2.7)), and time-based relevance model [15, 32] (see Equation (2.8)), which are respectively designated as RM, EXRM, and TBRM. The parameters of these PRF models are tuned. The parameter tuning and pre-process are discussed in Section 5.4.1 and 5.4.2. Figure 5.1 shows the bar plots of the difference in average precision of existing relevance models (RM, EXRM, and TBRM) over initial search results (LM) using 108 search topics for TREC 2011 and 2012 microblog track. Results showed that all PRF methods improved search performance for many topics, but simultaneously they decrease for several topics. The results imply that we must estimate more accurate temporal and lexical evidence for maintaining PRF performance and to improve microblog retrieval simultaneously.

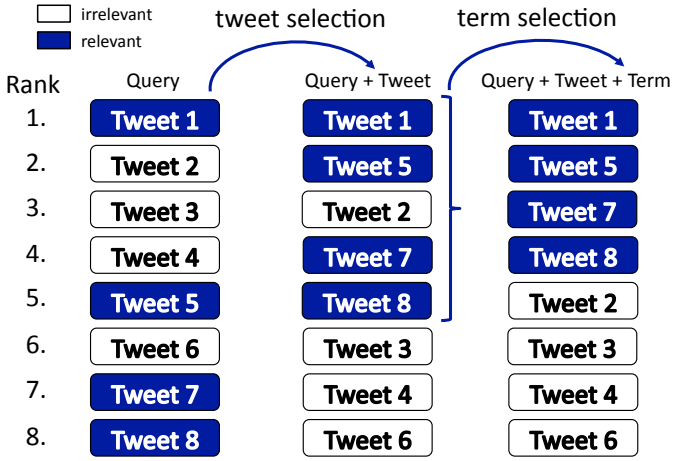

Figure 5.2: Overview of two-stage relevance feedback.

5.3 Proposed Method

To overcome the limitation of established PRF methods and to improve retrieval further, we propose two-stage relevance feedback methods. They consist of tweet selection feedback (TSF) and query-document dependent temporal relevance model. We describe an overview of our approach in Figure 5.2. For the former, we only select a single relevant tweet among initial search results and re-retrieve tweets using the selected tweet as expansion words of a new query. For the latter, we apply query-document dependent temporal query expansion method to the re-retrieved documents, which almost all include relevant tweets at the top. The following sections show details of the respective methods.

5.3.1 Tweet Selection Feedback

The first relevance feedback uses a selected tweet from the initial search results. We assume that the relevant tweet selected by users is a good indicator to retrieve relevant tweets to a given query because the relevant tweet generally includes good topic-related words. Using the selected tweet as expansion word for re-retrieving documents, we can obtain relevant tweets similar to the selected tweet at the top.

Additionally, we observed that the top ranked tweets retrieved at the top by a standard search engine with default settings (LM in our case) are often relevant, so that users can easily detect at least a relevant document from top ranked documents. To see the initial search performance, we define the proportion of search topics that retrieve at least a single relevant document among the top M' documents. We

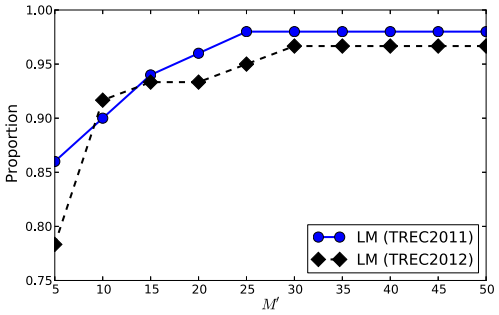

Figure 5.3: Proportion that at least one relevant document is contained among initial search results across different values of the cut off parameter M'.

have

$$\frac{1}{N'}\sum_{i=1}^{N'}\psi(P_i@M')$$

where $\psi(\cdot)$ is a function $\psi(x) = 1$ if $x > 0$; otherwise, $\psi(x) = 0$. N' is the number of topics used and $P_i@M'$ is the value of precision at M' for the i-th topic. Figure 5.3 presents the proportion across several cut off parameters M' using TREC 2011 and 2012 microblog track topics. Results show that users can find relevant tweets at the top without much effort in the case of many TREC search topics. For example, the proportion of finding at least one relevant document among the top 30 is more than 0.95 in both datasets. Furthermore, users can read many tweets quickly because the length of the tweet content is limited to 140 characters. Consequently, users can readily detect a relevant tweet without much effort.

5.3.2 Query-Document Dependent Temporal Relevance Model

In this section, we introduce the query-document dependent temporal relevance model. We assume that the search results by tweet selection feedback rank many relevant documents at the top, which contains more accurate word and temporal distributions than by initial search. To use the improved pseudo-relevance information effectively, we propose a novel relevance feedback approach using lexical and temporal evidence.

We rely mainly on the notion of Dakka et al. [16] and Efron and Golovchinsky [20] for time-sensitive language modeling frameworks and also use a document expansion approach proposed by Efron et al. [22] to capture document-dependent temporal variation. We explain the relevance model step-by-step. First, we decompose a document part D in $P(w|Q)$ into the lexical word in document D_w and

temporal information of document D_t following Dakka et al. [16],

$$\begin{aligned} P(w|Q) &= \sum_{D \in \mathcal{R}} P(w, D|Q) \\ &= \sum_{D \in \mathcal{R}} P(w, D_w, D_t|Q) \\ &= \sum_{D \in \mathcal{R}} P(w, D_w|D_t, Q) P(D_t|Q). \end{aligned}$$

Then, following Efron and Golovchinsky [20]'s work, we applied the simple assumption that the temporal relevance of D_t is independent of the document's content, D_w, and drop D_t from the conditional probability in Equation (5.1). Moreover, we assume that the given query consisting of query words in Q and the words w in pseudo-relevant documents are sampled identically and independently from a uni-gram distribution of $\mathcal{R}$. Therefore, we have

$$\begin{aligned} P(w|Q) &= \sum_{D \in \mathcal{R}} P(w, D_w|Q) P(D_t|Q) \\ &\propto \sum_{D \in \mathcal{R}} \underbrace{P(w|D_w) P(Q|D_w) P(D_w)}_{\textbf{Lexical}} \underbrace{P(D_t|Q)}_{\textbf{Temporal}} \\ &= \sum_{D \in \mathcal{R}} P(w|D_w) \prod_{i}^{|Q|} P(q_i|D_w) P(D_t|Q), \end{aligned}$$

where $P(D_t|Q)$ is the query-dependent document generation probability from a temporal perspective. We designate $P(D_t|Q)$ as *temporal evidence*. However, $P(w|D_w)P(Q|D_w)P(D_w)$ is equal to a factor of the standard relevance feedback model (see Equation (2.3)). We designate this as *lexical evidence*. In Equation (5.2), we assume that the prior probability over documents from a lexical perspective, $P(D_w)$, is uniform. Equation (5.2) is the weighted sum of query-dependent lexical evidence by query-dependent temporal evidence with respect to each document.

Ideally, the probability $P(D_t|Q)$ becomes high when the query Q and the document D share a similar temporal property, so that we quantify this temporal property as the distance between two temporal models of Q and D using the notion of temporal profile [29]. Borrowing the idea of temporal profile in Equation (2.9), we define the temporal models of Q and D as $P(t|Q)$ and $P(t|Q_D)$, respectively, where Q_D is the pseudo-query of D submitted to search engines as a query based on the idea of Efron et al. [22]. Using $P(t|Q_D)$, we can capture document-dependent temporal variation. In addition, we apply background smoothing to both temporal models and then smooth them with the model for

adjacent days following previous works [16, 29]. Additionally, we assume that the distance between two temporal models approximately follows an exponential distribution because the documents retrieved by Q_D share more similar temporal property with the documents retrieved by Q than unobserved documents. We define the probability of query-dependent document's temporal evidence as

$$P(D_t|Q) \propto P(X > d) = e^{-\gamma d},$$

where d is the distance of two temporal models between $P(t|Q)$ and $P(t|Q_D)$ and γ is a rate parameter of exponential distribution. Moreover, past works [19, 20] have shown that incorporating query-dependent recency is effective for improving microblog search. Therefore, we design the rate parameter as automatically changing in response to each query's temporal property, as

$$\gamma = 1 - \sum_{t \in \mathcal{T}_Q} P(t|Q). \tag{5.4}$$

where $\mathcal{T}_Q = \{t \in \mathcal{T} : t_Q - t < \alpha\}$, $\mathcal{T}$ is a time range in a collection (days in our case), t_Q denotes a query-time of query Q, and α is a hyper-parameter that controls the impact of topic-recency. The probability γ denotes the value of complementary cumulative distribution function of temporal model until α days before the topic's query-time. If the temporal profile of a given query ranks many documents generated at around its query-time at the top, then the probability γ is low. However, the probability is high if those document time-stamps are far from the query-time.

We assume that similar temporal models should share similar temporal property (e.g. temporal variation). Therefore, we compare two temporal models using the Bhattacharyya coefficient,

$$\mathcal{B}(Q, D) = \sum_{t \in \mathcal{T}} \sqrt{P(t|Q)P(t|Q_D)}.$$

This comparison provides a similarity score between 0 and 1. Similar methods have been used to compare two associated language models using the Bhattacharyya coefficient [18]. Using the Bhattacharyya coefficient, we can obtain the distance between two temporal models as

$$d = -\ln \mathcal{B}(Q, D).$$

This is called Bhattacharyya distance. When we substitute Equation (5.6) into Equation (5.3), we have the following final equation:

$$P(D_t|Q) \propto \{\mathcal{B}(Q, D)\}^{\gamma}.$$

This probability $P(D_t|Q)$ becomes high when $P(t|Q)$ and $P(t|Q_D)$ are similar (i.e. Bhattacharyya coefficient is high). The increase of $P(D_t|Q)$ approaches linear increase when γ is high; in other words, a given topic indicates an old event. However, $P(D_t|Q)$ rapidly increases when a given topic indicates a recent event (i.e. γ is low).

5.4 Evaluation

5.4.1 Experimental Setup

We evaluate our proposed methods using the test collection for the TREC 2011 and 2012 microblog track (Tweets2011 corpus). In addition, relevance judgment is applied to the whole tweet set of each topic. The relevance levels are categorized into irrelevant (labeled 0), minimally relevant (labeled 1), and highly relevant (labeled 2). We separately evaluate our methods as *allrel* and *highrel*, where allrel considers both minimally relevant and highly relevant tweets as relevant and highrel considers only highly relevant tweets as relevant.

We indexed tweets posted before the specific time associated with each topic by the Indri search engine with the following setting. All queries and tweets are stemmed using the Krovetz stemmer without stop-word removal. They are case-insensitive. This index was created to simulate a realistic real-time search setting, where no future information is available when a query is issued. We built an index for each query. In our experiments, we used the titles of TREC topics numbered 1–50 and 51–110[3] as test queries, which are the official queries in the TREC 2011 and 2012 microblog track, respectively. Additionally, we used 33 topics at TREC 2011 and 56 topics at TREC 2012, and obtained highly relevant tweets for highrel.

For retrieving documents, we used a basic query likelihood model with Dirichlet smoothing [83] (we set smoothing parameter $\mu = 2500$ similar to Efron's work [22]) implemented by the Indri search engine [73] as the language model for IR (LM) and all PRF used this LM as initial search results. We filtered out all non-English retrieved tweets using a language detector with infinity-gram, called ldig[4]. The retweets[5] were regarded as irrelevant for evaluation in the TREC microblog track [61, 72]; however, we used retweets except in a final ranking of tweets because a set of retweets is a good source might contain topic-related words for

[3]The topic numbered MB050 and MB076 has no minimally or highly relevant tweets. Therefore, we did not use them for our experiments.

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

[5]Tweets re-posted by another user to share information with other users

improving Twitter search performance [15]. In accordance with the track's guidelines, all tweets with http status codes of 301, 302, 403, and 404 and all retweets contain the string "RT" at the beginning of the tweet were removed from the final ranking. Finally, we used the top 1000 results for evaluation.

5.4.2 Baselines

Our approach first conducts tweet selection feedback (TSF) described in Section 5.3.1. We automatically select relevant tweets from initial search results among top L tweets for TSF by each topic. We set L to 30 based on a preliminary experiment. In Section 5.4.4, we show that the performance is not sensitive to the choice of L when L is sufficiently large (e.g. $L \geq 30$). The selected relevant tweets are minimally or highly relevant tweets. When multiple relevant tweets exist in initial search results, we use only a single relevant tweet that contains more words in it than others. We assume that users prefer long tweets. If relevant tweets do not exist among initial search results, we use the original user query for tweet selection feedback. All selected tweets were stopped using Indri's stop words list with URL and mention (e.g. @trecmicroblog) removal. In the new query, the selected tweet and the original query were weighted as $1 : 1$ for each method using TSF. After tweet selection feedback, we conduct the proposed query expansion method based on a query-and-document dependent temporal relevance model (QDRM). For QDRM, we produce a temporal profile consisting of the top N tweets, which were retrieved using a document among initial search results as a pseudo-query. These pseudo-queries were also pre-processed in the same mode as tweets used for TSF. We denote the combination of TSF and QDRM as TSF + QDRM.

To assess our proposed methods, TSF and TSF + QDRM, we also prepared several baseline methods. Our first baseline, RM, uses standard relevance feedback using only lexical evidence [40]. This can be compared with TSF + RM which uses tweet selection feedback before the pseudo-relevance feedback RM. QDRM differs from RM in that RM does not consider temporal evidence. Actually, QDRM is equal to RM when we set γ in QDRM to 0 (see Equations (5.2) and (5.3)). Our second baseline, EXRM uses relevance feedback using exponential distribution to prior probability for relevance model [42]. EXRM does not consider query-dependent recency and temporal variation compared to QDRM. We also prepare TSF + EXRM, which is a combination of TSF and EXRM to assess the effect of tweet selection feedback for the recency-based method. Finally, our third baseline is a time-based relevance model, TBRM, that incorporates lexical evidence and query-dependent temporal variation into its relevance model. However, it ignores recency and document-dependent temporal variation. We compare this model and its tweet selection extension, TSF + TBRM, to our

QDRM that uses both lexical and temporal evidence with query-dependent recency. RM, EXRM, and TBRM are strong baselines in our experiments.

For all query expansion methods, we select candidate words among the top M tweets retrieved by the original query after removing the uniform resource locators (URLs), and user names starting with '@' or special characters (!, @, #, ', ", etc.). All query words, candidate words, and tweets are decapitalized. The candidate words include no stop-words prepared in the Indri search engine. Then, we select K words among candidate words in descending order of the probability $P(w|Q)$, respectively. The selected words contain no original query word, but might contain words of the selected tweet in the case of using TSF. Finally, we combined the expanded words of PRF and the original query (or the combination of the original query and the selected tweet) as an expanded query; they were weighted with 1 : 1.

For QDRM and EXRM, we tune parameters: the length of temporal profile (i.e. N), the hyper-parameter (i.e. α), and the rate parameter (i.e. r). For all methods, we also tune their parameters: the number of pseudo-relevance feedback documents (i.e. M) and the number of expansion words (i.e. K). Values of the model parameters are optimized for best performance precision at 30 on training data, which is the official measure in TREC 2011 microblog track. For example, we tune parameters of the IR model using TREC 2012 microblog track dataset and test it with TREC 2011 microblog dataset. However, we trained the model using the TREC 2012 dataset and test it on the TREC 2011 dataset. Results show that the parameter N in the proposed QDRM set to be 10 is better for both datasets. The sensitivity of other important parameters such as L in TSF and the recency control parameter α of QDRM is discussed in the next section.

5.4.3 Evaluation Measure

To evaluate retrieval effectiveness, we used precision at 10 and 30 (P@10, P@30, respectively), average precision (AP), and normalized discounted cumulative gain (nDCG) [28], nDCG considers graded relevance. In the TREC 2012 microblog track, "highly relevant" tweets are the required level of relevance. We discuss the statistical significance of results obtained using a two-sided Fisher's randomization test [71] throughout this chapter.

Table 5.1: Performance comparison of the proposed methods and baselines for allrel documents.

	TREC 2011			
Method	AP	nDCG@10	P@10	P@30
LM	0.3571	0.5301	0.4755	0.4143
RM	0.4063_{l}	0.5616	0.5673_{l}	0.4741_{l}
EXRM	0.4204_{lr}	0.5725	0.5816_{l}	0.4762_{l}
TBRM	0.4020	0.5573	0.5673_{l}	0.4728_{l}
QDRM	0.4206_{l}	0.5843	0.5735_{l}	0.4721_{l}
TSF + LM	$0.5040^{\blacktriangle}$	$\mathbf{0.6956}^{\blacktriangle}$	$0.6388^{\blacktriangle}$	$0.4966^{\blacktriangle}$
TSF + RM	$0.5287^{\blacktriangle}$	$0.6730^{\blacktriangle}$	$0.6327^{\triangle}$	$0.5224_{l'}$
TSF + EXRM	$0.5328^{\blacktriangle}$	$0.6814^{\blacktriangle}$	$0.6449^{\triangle}$	$0.5218^{\triangle}_{l'}$
TSF + TBRM	$0.5174^{\blacktriangle}$	$0.6745^{\blacktriangle}$	$0.6429^{\blacktriangle}$	0.5177
TSF + QDRM	$\mathbf{0.5384}^{\blacktriangle}_{l'}$	$0.6843^{\blacktriangle}$	$\mathbf{0.6571}^{\blacktriangle}_{r'}$	$\mathbf{0.5354}^{\blacktriangle}_{l'}$
	TREC 2012			
LM	0.2408	0.4177	0.4814	0.3847
RM	0.3024_{l}	0.4592_{l}	0.5475_{l}	0.4503_{l}
EXRM	0.3025_{l}	0.4663_{l}	0.5492_{l}	0.4520_{l}
TBRM	0.3139_{l}	0.4826_{l}	0.5610_{l}	0.4644_{lq}
QDRM	0.3039_{l}	0.4760_{l}	0.5542_{l}	0.4441_{l}
TSF + LM	$0.3198^{\blacktriangle}$	$0.5309^{\blacktriangle}$	$0.5763^{\blacktriangle}$	$0.4559^{\blacktriangle}$
TSF + RM	$0.3475^{\triangle}_{l'}$	$0.5352^{\triangle}$	$0.6068^{\triangle}$	0.4785
TSF + EXRM	$0.3476^{\triangle}_{l't'}$	0.5329	$0.6068^{\triangle}$	0.4797
TSF + TBRM	$0.3415_{l'}$	0.5331	0.6051	0.4763
TSF + QDRM	$\mathbf{0.3584}^{\blacktriangle}_{l'r'e't'}$	$\mathbf{0.5552}^{\blacktriangle}_{r'e'}$	$\mathbf{0.6220}^{\blacktriangle}$	$\mathbf{0.4910}^{\blacktriangle}_{l'}$

5.4.4 Experimental Results

Overall Results

Table 5.1 shows the P@10, P@30, AP, and nDCG performances of 10 methods with statistical significance test results for allrel documents. Table 5.2 shows the P@30 and AP performances for highly relevant documents. Significant improvements by tweet selection feedback (TSF) are denoted with $\triangle$ and $\blacktriangle$, respectively, for significance probabilities $p < 0.05$ and $p < 0.01$. In addition, among methods without the use of TSF, the subscript l, r, e, t, and q respectively indicate statistically significant improvements ($p < 0.05$) over LM, RM, EXRM, TBRM, and QDRM. Moreover, among methods using TSF, the subscripts l', r', e', t', and q' respectively indicate statistically significant improvements ($p < 0.05$) over TSF

Table 5.2: Performance comparison of the proposed method and baselines for highrel documents of TREC 2011 and 2012 datasets.

	TREC 2011		TREC 2012	
Method	AP	P@30	AP	P@30
LM	0.2747	0.1293	0.1766	0.1976
RM	0.2499	0.1374	0.2258_{l}	0.2494_{l}
EXRM	0.2710_{t}	0.1465_{r}	0.2270_{l}	0.2548_{l}
TBRM	0.2404	0.1374	0.2314_{l}	0.2583_{l}
QDRM	0.2911	0.1424	0.2293_{l}	0.2500_{l}
TSF + LM	$0.3461^{\triangle}$	0.1566	$0.2180^{\blacktriangle}$	$0.2387^{\blacktriangle}$
TSF + RM	$0.3508^{\triangle}$	$0.1727^{\triangle}$	0.2358	0.2595
TSF + EXRM	0.3476	0.1747	0.2358	0.2613
TSF + TBRM	$0.3365^{\triangle}$	0.1717	0.2325	0.2542
TSF + QDRM	$\mathbf{0.3619}_{l'}$	$\mathbf{0.1758}^{\triangle}$	$\mathbf{0.2389}_{l'}$	$\mathbf{0.2649}_{l'}$

+ LM, TSF + RM, TSF + EXRM, TSF + TBRM, and TSF + QDRM. The best result per column is marked in bold typeface.

It is apparent that QDRM markedly outperforms the initial search LM on most measures across both datasets, similarly to other relevance feedback approaches RM, EXRM, and TBRM with statistical significance. Moreover, QDRM outperformed the standard relevance model RM in terms of most evaluation measures across both datasets similar to other time-based relevance feedback methods EXRM and TBRM, which suggests that temporal evidence (recency or temporal variation) is important for microblog search. However, none of these differences is statistically significant except between RM and EXRM on AP.

When using tweet selection feedback, TSF + LM markedly outperformed LM in terms of all measures across both datasets with statistical significance, which suggests that the simple query expansion method using a selected relevant tweet as expansion words is considerably effective. Furthermore, relevance feedback approaches after TSF outperformed relevance feedback without using TSF in terms of all measures. For all using TSF, the differences in AP, nDCG@10, and P@10 in the TREC 2011 dataset were statistically significant. Important points include the fact that TSF + QDRM markedly outperformed QDRM with regard to all evaluation measures across both datasets with statistical significance. For both datasets, TSF + QDRM outperformed other PRF methods using TSF: TSF + RM, TSF + EXRM, and TSF + TBRM. Particularly the difference in average precision on the TREC 2012 dataset is statistically significant. Results suggest that tweet selection feedback is useful for PRF methods and that incorporating query-dependent lexical and temporal evidence by each document is considerably

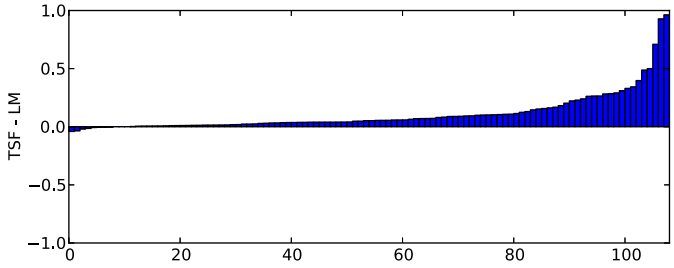

Figure 5.4: Difference in average precision between TSF and LM using the TREC 2011 and 2012 microblog track topics.

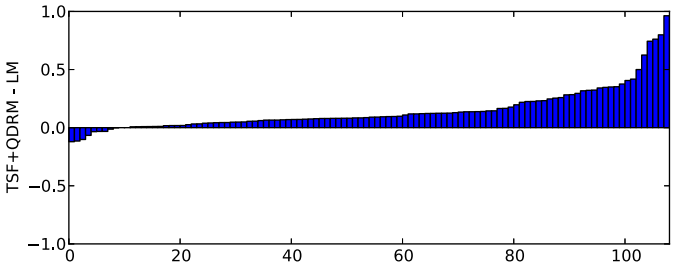

Figure 5.5: Difference in average precision between TSF + QDRM and LM using the TREC 2011 and 2012 microblog track topics.

effective when using improved search results by tweet selection feedback.

From Table 5.2, it is also apparent that PRF using TSF is effective for improving retrieval performance when searching highly relevant documents. In this case, TSF + QDRM outperformed other methods in all evaluation measures across both datasets. For further improvement of search performance with regard to highly relevant documents, we must consider external web-contents corresponding to URLs in a tweet, which significantly affect the retrieval performance of highly relevant tweets [43].

Overall Results

We underscore the effectiveness of tweet selection feedback (TSF) comparing to initial search results (LM) in Figure 5.4. The bar plot shows the difference in average precision between LM and TSF on a query-by-query basis. Compared to relevance feedback methods without tweet selection feedback shown in Figure 5.1, TSF not only significantly improved search results over the initial search (see Table 5.1); it also improved the search performance of each topic without decreasing search performance over almost all topics. For example, Table 5.3

Table 5.3: Improved and decreased percentages of the values of mean average precision (MAP [%]) and the number of topics (#) by pseudo-relevance feedback methods over the initial search using the TREC 2011 and 2012 topics.

	Improved		Decreased	
Method	MAP [%]	#	MAP [%]	#
RM	51.5 (93.9)	87	-37.0 (24.8)	20
EXRM	55.4 (96.2)	87	-30.5 (26.6)	20
TBRM	64.4 (107.2)	81	-29.5 (24.9)	25
QDRM	46.3 (65.1)	86	-21.8 (18.2)	18
TSF + LM	122.4 (314.9)	97	-4.8 (4.4)	8
TSF + RM	161.4 (350.6)	92	-25.4 (17.0)	14
TSF + EXRM	162.2 (348.2)	92	-25.5 (17.0)	14
TSF + TBRM	160.6 (350.9)	91	-28.7 (20.1)	16
TSF + QDRM	153.7 (337.6)	97	-26.3 (19.0)	9

shows that TSF + LM improved results for about 97 topics, and decreased results for about 8 topics, whereas the results of relevance feedback methods without the use of TSF (RM, EXRM, TBRM, and QDRM) improved about 81–87 topics and decreased about 18–20 topics.

In addition, Figure 5.5 shows the results for relevance feedback after tweet selection (TSF + QDRM). Table 5.3 shows that TSF + RM, TSF + EXRM, TSF + TBRM, similarly to TSF + QDRM also improve retrieval performance for almost all topics without decreasing search performance compared to RM, EXRM and TBRM, which suggests that tweet selection feedback combined with PRF is effective to improve retrieval performance steadily. Particularly, we found that TSF + QDRM effectively uses search results refined by tweet selection feedback compared to other relevance feedback methods.

Parameter Sensitivity

In our experiments, we selected a longest tweet among the top 30 tweets retrieved by LM (i.e. $L = 30$) and combined it with an original query as a new query for tweet selection feedback. We demonstrate in Figure 5.6 how the value of mean average precision (MAP) of TSF changes with different L parameters. Results showed that the performances of TSF + LM increase until $L = 30$, and become insensitive to L when L is large (e.g. $L \geq 30$) on both datasets. Those results suggest that the top ranked 30 tweets tend to contain a relevant tweet that can improve the retrieval performance via TSF, so that microblog users should read the top 30 tweets and select only a single relevant tweet among them when searching

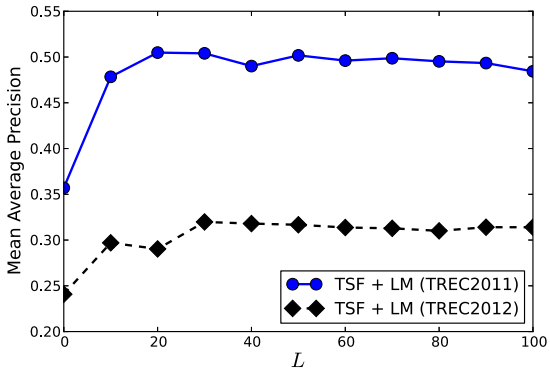

Figure 5.6: Sensitivity to the number of top retrieved tweets L used for tweet selection feedback. The x-axis shows the value of L. The y-axis shows the value of mean average precision over the TREC 2011 and 2012 microblog track topics, respectively.

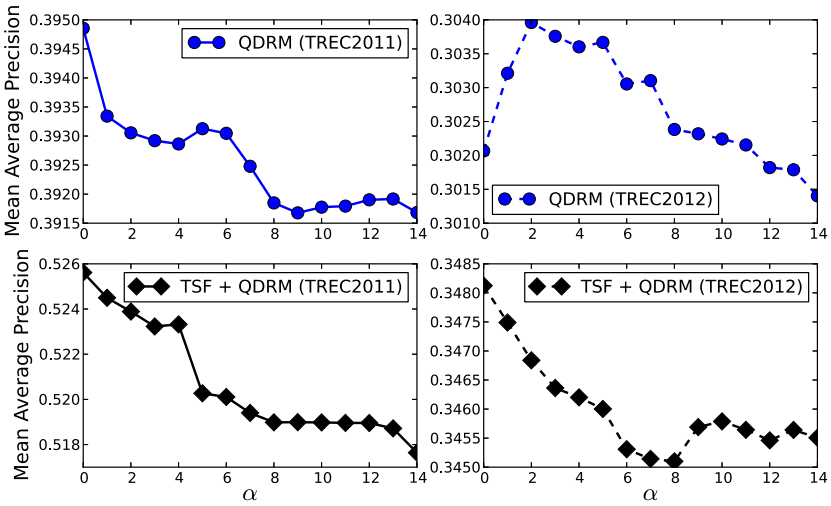

Figure 5.7: Sensitivity to the recency control parameter α used in QDRM over QDRM and TSF + QDRM at TREC 2011 (left-top and bottom) and QDRM and TSF + QDRM at TREC 2012 (right-top and bottom). The x-axis shows the values of α. The y-axis shows the value of mean average precision.

the Tweets2011 corpus effectively.

We also show the parameter sensitivity of α in QDRM. The parameter α controls the degree of the recency parameter over topics. Figure 5.7 shows the MAP values of QDRM and TSF + QDRM for $L = 30$, $M = 100$, $N = 10$, and $K = 20$

across different α values. It is readily apparent that the performance of QDRM and TSF + QDRM on both datasets is sharply decreasing when using large α values. Large α tempers the impact of temporal evidence because γ value tends to approach 0 (see Equations (5.4) and (5.7)). The results suggest that query and document-dependent temporal evidence in QDRM is working. The optimal value of QDRM on TREC 2012 dataset is $\alpha = 2$, which indicates the effectiveness of recency. However, the difference of MAP values is slight. We assumed that this robustness results from the short time span of the Tweets2011 corpus (about two weeks). It was also described earlier in the past work [22]. The optimal α values of TSF + QDRM on both datasets were 0, which means considering only query and document's temporal variation in temporal evidence ignoring recency is effect. We assumed that TSF was able to bring more accurate temporal distributions, so that the recency effect of QDRM was vanishingly small.

Temporal Analysis

We evaluate TSF from a temporal perspective. To demonstrate the improvement of estimation of temporal evidence, we compared the time-stamp distribution of relevant documents to that of the first 20 documents retrieved by a simple query likelihood model (LM) and tweet selection feedback (TSF + LM), respectively. Bhattacharyya coefficient is used as the similarity between two time-stamp distributions of retrieved documents. The higher value of the Bhattacharyya coefficient means that the IR system precisely estimates topic-related temporal evidence. Figure 5.8 shows the Bhattacharyya coefficients of LM and TSF + LM against relevant documents. To test the difference of the Bhattacharyya coefficient between LM and TSF + LM, we use two-sided Wilcoxon matched-pairs signed-ranks test with $p < 0.05$. Results show that the Bhattacharyya coefficient improved after TSF and the variance was smaller than LM. For example, the coefficient of TSF on TREC 2011 dataset significantly outperformed that of LM (from 0.7748 ± 0.1756 to 0.8071 ± 0.1566), but when using the TREC 2012 dataset, the difference is not statistically significant (from 0.7822 ± 0.1361 to 0.7988 ± 0.1112). The coefficient values of LM and TSF on both datasets (ALL) are 0.7789 ± 0.1553 and 0.8026 ± 0.1338, respectively. The difference is statistically significant. The point is that we can predict accurate temporal evidence using TSF.

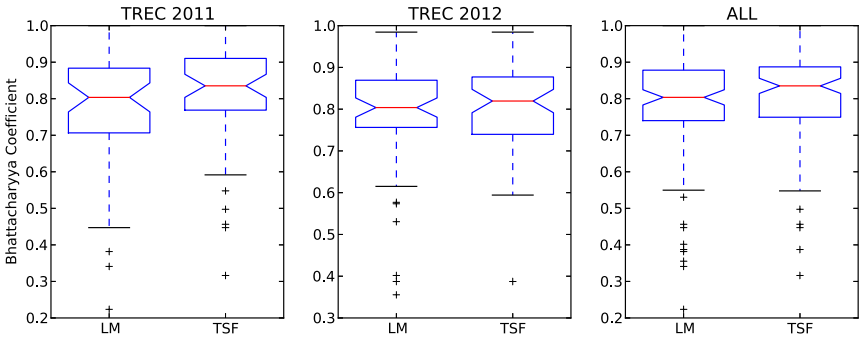

Figure 5.8: Bhattacharyya coefficient between temporal profiles of LM and TSF using the TREC 2011 and 2012 datasets.

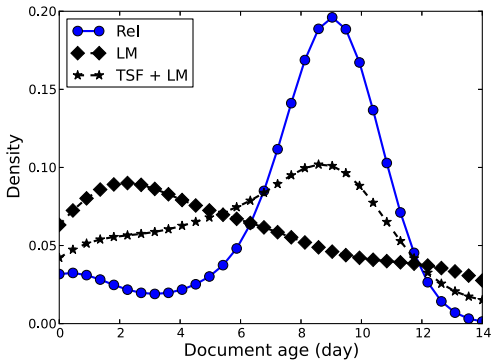

Figure 5.9: Temporal variations of a topic numbered MB042. The x-axis shows the document age from the query-time when query was issued to document time-stamp. The y-axis shows the kernel-estimated probability density for the document age. The blue line (Rel) shows estimates for relevant documents. Black lines (LM and TSF + LM) respectively show estimates of the top 30 retrieved documents by LM and TSF. High density indicates the period during which the topic was described actively.

Query Analysis

In Table 5.4, we display candidate words of query expansion for the topic *"Holland Iran envoy recall"* (MB042[6]) in RM, in QDRM, in TSF + RM, and in TSF + QDRM, showing improved results with TSF + QDRM. It is apparent that more topic-related words such as "dutch", "bahrami", and "iranian" appear in our ap-

[6]The news that "Dutch government is recalling its Tehran ambassador for consultations over the burial of executed Dutch–Iranian Sahra Bahram" was reported by BBC News on 7 February 2011.

Table 5.4: Expanded words for a topic numbered MB042: *"Holland Iran envoy recall"*.

RM		QDRM		TSF + RM		TSF + QDRM	
word	$P(w\|Q)$	word	$P(w\|Q)$	word	$P(w\|Q)$	word	$P(w\|Q)$
mubarak	0.057	mubarak	0.053	dutch	0.018	dutch	0.078
iranian	0.046	egypt	0.033	iranian	0.017	iranian	0.044
dutch	0.040	obama	0.031	bahrami	0.013	woman	0.032
says	0.036	rt	0.019	rt	0.012	bahrami	0.020
special	0.029	special	0.016	zahra	0.011	mubarak	0.019
row	0.028	stay	0.015	iranelection	0.009	zahra	0.017
stay	0.024	says	0.014	mubarak	0.009	drug	0.015
un	0.021	wisner	0.014	execution	0.008	hanging	0.014
news	0.020	jan25	0.011	woman	0.007	rt	0.013
egypt	0.018	frank	0.011	egypt	0.006	government	0.012

proaches TSF + RM and TSF + QDRM. That is true because TSF selected *"Breaking: Dutch recall ambassador from #Iran over execution of Dutch–Iranian Zahra Bahrami, summon Iran Ambassador"* as a relevant document and used it for tweet selection feedback and refined lexical and temporal evidence for PRF. In addition, Figure 5.9 shows the kernel density estimate of the document age of topic MB042 using relevant tweets and top search results obtained using LM and TSF. From this figure, it is apparent that temporal variation approaches relevant estimates using TSF. Moreover, the word weights against a query, $P(w|Q)$, of topic-related words of TSF + QDRM are larger than that of TSF + RM. Results show that the average precision values of RM, QDRM, TSF + RM, and TSF + QDRM improved over the initial search LM (from 0.0490 to 0.0815, 0.0427, 0.8412, and 0.8478, respectively).

However, regarding *"Australian Open Djokovic vs. Murray"* (MB071), the average precision of relevance methods RM, QDRM improved over LM (from 0.5704 to 0.5809 and 0.6039, respectively) although that of TSF + RM, and TSF + QDRM decreased (from 0.5704 to 0.4450 and 0.4496, respectively). That is true because a tweet selected by TSF about this topic, *"Tomorrow is the Australian open tennis final for men, Andy Murray vs. Navok Djokovic Who's gonna win?? I'm a Murray fan so I say GO MURRAY!!"*, contains numerous topic-unrelated words.

5.5 Summary

In this chapter, we proposed two-stage relevance feedback approaches for microblog search using tweet selection feedback and query-document dependent

temporal relevance feedback methods. Our two-stage relevance feedback considerably improved retrieval performance with minimum user interaction. First, the user selects only one relevant tweet among top ranked initial search results and combines it with an original user query for tweet selection feedback (TSF), where the combined query is used for re-retrieving documents. Second, to improve search results further, a query-dependent relevance model QDRM is applied to top ranked re-retrieved documents.

TSF is a simple and effective approach to overcome the vocabulary mismatching problem and to improve microblog retrieval performance. Microblog documents are very short and tend to mention a single topic. TSF succeeds in exploiting the microblog feature. The user can quickly read and can readily select a relevant document among top re-retrieved search results that contain good words. A set of document time-stamps indicates the topic-related time. Using improved top search results for relevance feedback, we were able to improve search results using our proposed QDRM, which combines lexical and query-document dependent temporal evidence. Our two-stage relevance feedback framework can plug in any PRF method after TSF. We evaluated our approach using the Tweets2011 corpus with TREC 2011 and 2012 microblog datasets. The experimentally obtained results indicate that TSF markedly improves retrieval performance without decreasing over almost all queries. In addition, the proposed PRF method, QDRM, further considerably improved microblog search performance compared to established PRF methods.

As described in chapter 3, 4, and 5, time-aware information retrieval methods are effective to retrieve time-stamped documents; however, they do not consider how important the document is. In social networks, important and influential users tend to create high quality documents and their authority constantly changes over time. In the next chapter, we introduce time-aware object ranking method to consider importance of users in a social network.

Chapter 6

Time-Aware Object Ranking

6.1 Introduction

Object ranking plays an important role in the field of information retrieval. Particularly PageRank [11] and HITS [34] algorithms are remarkable techniques to rank objects such as documents or users. These algorithms typically rely on the existing relations between objects in networks. However, such relations change dynamically over time, as demonstrated in Section 2.3.2. Existing approaches that rely on the current network would have been limited to prediction of the future importance of objects. Therefore, we explore the question of how to predict the future importance of objects in social networks that change dynamically over time.

To address this question, in this chapter, we introduce the object ranking method, which identifies influential or important objects in future networks. We define

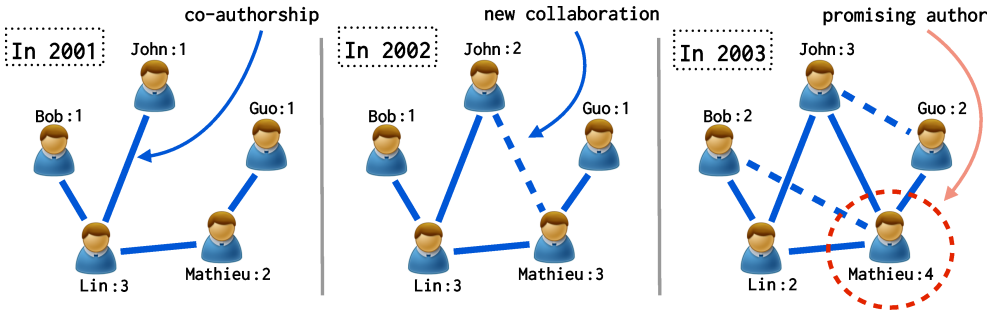

Figure 6.1: Example of a promising object in an evolutional network.

them as "promising objects." Figure 6.1 shows an example of a promising object in an evolutionary network, which represents co-authorship networks in 2001, 2002, and 2003. The number after the author's name is the number of collaborators in each network, which is called *degree centrality*. As this figure shows, Mathieu has more collaborators than others in 2003 although he had a few collaborators in 2001, so Mathieu is a promising author in this co-authorship network. To identify Mathieu as a promising author, it is necessary to predict the appearance of new collaborations. Consequently, our proposed method first predicts future links between objects using supervised link prediction. In addition, to predict importance of objects precisely, we use a learning-to-rank method. To demonstrate the effectiveness of our proposed method, we conducted experiments on a real social network dataset.

The remainder of the chapter is organized as follows. In Section 6.2 we describe the relation between link prediction and the ranking of objects, which changes dynamically over time. Section 6.3 presents details of the proposed method to detect promising objects through link prediction and object ranking. Experimental results obtained using arXiv (hep-th) [75] dataset are presented in Section 6.4. Finally, Section 6.5 gives a summary of this chapter.

6.2 Link Prediction and Object Ranking

The proposed method predicts future links for detecting promising objects in social networks. In this section, we clarify the relation between the link prediction and the importance of object that changes dynamically over time.

6.2.1 Link-Based Object Ranking

This section presents the ability to predict object ranking using the appearance of true and false links. The true links are those that actually appear in a future network, whereas false links do not. As shown in Figure 6.1, the appearance of new links affects object ranking ordered by network centralities. Therefore, we validate the precise prediction of true links that lead to correct object ranking. We use arXiv (hep-th) datasets described in Section 2.3.2.

The following procedure is used for prediction:

1. calculate the percentage β of true links that appeared during 1995–2003
2. select $\beta \times k$ true links at random, where k denotes the number of links added to an initial network

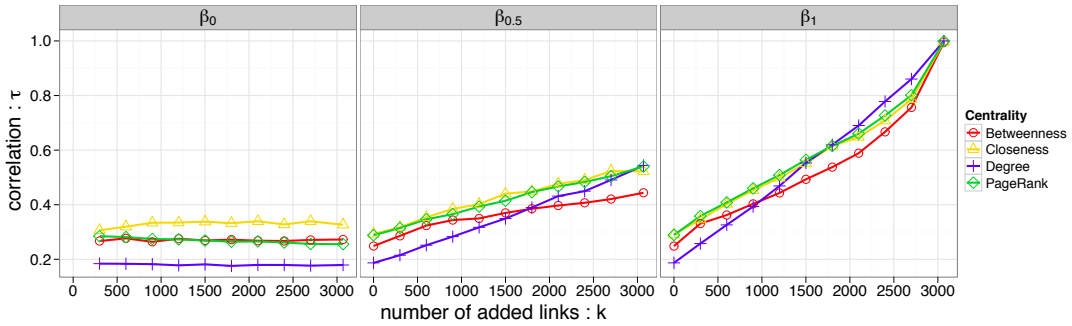

Figure 6.2: Rank correlation between the true ranking and the predicted one with $\beta = 0$ (left), $\beta = 0.5$ (center), and $\beta = 1$ (right).

3. add the selected links to an initial network in 1994 and predict a future network in 2003
4. calculate network centralities of each object in the initial network and the predicted one
5. rank objects by their network centralities
6. calculate the rank correlation τ of top 100 objects
7. repeat the steps above 30 times and calculate the average τ

The high β values in link prediction denote that we can predict true links in the future network precisely. Figure 6.2 shows rank correlations between the true ranking and the predicted one. As the figure shows, the number of added links k has an almost linear effect on τ across several β values. The result suggests that prediction of true links leads to the ranking of objects correctly. In addition, when $\beta = 1$ (i.e., all added links are true links), link prediction can predict future object ranking precisely along with the number of added links.

6.2.2 Link Prediction with Object Importance

In Section 6.2.1, we described that the addition of true links to an initial network precisely predicts the future object ranking. However, the former link prediction method considers only the links; it does not assess the importance of objects in a social network. We assume that the links connected to important objects have the power to predict promising objects.

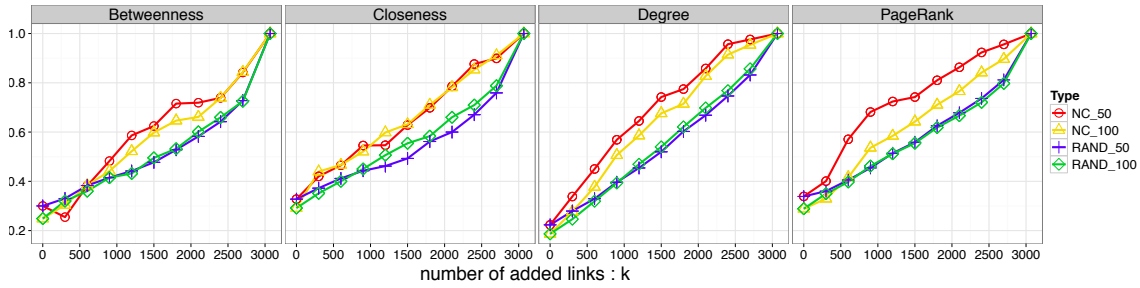

Figure 6.3: Rank correlation between the true ranking and the one predicted by adding links with network centralities.

To validate the idea, we use link prediction considering the importance of objects (network centrality in our case). The following procedure is used for prediction:

1. rank all true links by a network centrality of the object that connects to the links
2. add top k links to an initial network in 1994 and predict a future network in 2003
3. calculate network centralities of each object in the initial network and the predicted one
4. rank objects by their network centralities
5. calculate the rank correlation τ of top 100 objects
6. repeat the above steps 30 times and calculate the average τ

Figure 6.3 shows the rank correlation between the true ranking and the predicted one by adding links with network centralities. "RAND_50" and "RAND_100" indicate the results of link prediction respectively described in Section 6.2.1 using the top 50 and 100 objects. "NC_50" and "NC_100" respectively denote the results of link prediction with object importance using the top 50 and 100 objects. As the figure shows 6.3, we found that all rank correlations τ increase almost linearly as the number of added links increases. Particularly, the rank correlations of "NC_50" and "NC_100" outperformed "RAND_50" and "RAND_100" for all network centralities, which suggests that the importance of objects in a network is useful for predicting true links. In addition, the curve of "NC_50" grows more sharply than that of "NC_100", which suggests that top ranked objects are more important to predict promising objects in a future network.

6.3 Proposed method

6.3.1 Link Prediction

In this section, we present the link prediction method using machine learning approach. The link prediction in Section 6.2.1 showed that predicting true links engenders precise ranking objects in a future network. However, all true links are unknown. Consequently, for finding promising objects, we need to predict future links from a given network. In our study, we use the structure-based link prediction method, which uses the network structure as features for machine learning approaches. The feature is called a *link index*, which is defined as the network structure around a pair of objects.

Overall, we use the following link index types for link prediction.

- **Common neighbors (CN)**
  Common neighbors denote the number of objects which two objects have in common through one link. Using this index, Newman[59] showed the relation between the number of common neighbors of two objects and the probability that they will collaborate in the future. This index is defined as

  $$\mathrm{CN}(v^{(i)}, v^{(j)}) = |\Gamma(v^{(i)}) \cap \Gamma(v^{(j)})|,$$

  where the score of edge $<v^{(i)}, v^{(j)}>$ between two objects $v^{(i)}$ and $v^{(j)}$ and $\Gamma(v)$ is a set of neighbors in a network.

- **Jaccard's coefficient (JAC)**
  Jaccard's coefficient is the number of intersected objects divided by the number of coupled objects. This measure is sometimes used as a similarity metric in information retrieval [49]. The Jaccard's coefficient is defined as

  $$\mathrm{JAC}(v^{(i)}, v^{(j)}) = \frac{|\Gamma(v^{(i)}) \cap \Gamma(v^{(j)})|}{|\Gamma(v^{(i)}) \cup \Gamma(v^{(j)})|}.$$

- **Adamic/Adar (ADA)** [1]
  This index gives more weight to neighbors that are not shared with many others. Adamic/Adar is defined as

  $$\mathrm{ADA}(v^{(i)}, v^{(j)}) = \sum_{k \in |\Gamma(v^{(i)}) \cap \Gamma(v^{(j)})|} \frac{1}{\log |\Gamma(v^{(k)})|}.$$

- ***Katz*$_\beta$(KB)** [31]
  $Katz_\beta$ directly sums a collection of paths and is exponentially damped by length to count short paths more heavily. $Katz_\beta$ is defined as

$$\mathrm{KB}(v^{(i)}, v^{(j)}) = \sum_{l=1}^{\infty} |\mathrm{paths}^{(l)}_{v^{(i)}, v^{(j)}}|,$$

  where $\mathrm{paths}^{(l)}_{v^{(i)}, v^{(j)}}$ represents the number of paths from $v^{(i)}$ to $v^{(j)}$ with length l. If the length l is 1, then this metric represents the number of collaborations that $v^{(i)}$ and $v^{(j)}$ have.

- **Preferential Attachment (PA)** [4]
  Preferential attachment is based on a generative model of scale-free network. This metric assumes that the probabilities of a relation of $v^{(i)}$ and $v^{(j)}$ are respectively correlated with the product of the number of links of $v^{(i)}$ and y, $|\Gamma(v^{(i)})|$ and $|\Gamma(v^{(j)})|$, respectively. This index is defined as

$$\mathrm{PA}(v^{(i)}, v^{(j)}) = |\Gamma(v^{(i)})| \times |\Gamma(v^{(j)})|.$$

For predicting future links, we use a decision tree, C4.5 [65], as a link predictor. This link predictor uses the link index of all pairs of objects as features and uses the existence of links as class labels. The decision tree predicts future links with the probability of link appearance. We regard the predicted links that have high probability as new links. In addition, to overcome the imbalance problem [27] we use down-sizing and bagging methods. First, we randomly sample as many true links as false ones. Second, we produce several training datasets using sampled links. Then, we train decision trees with each dataset. Finally, we average the probabilities produced by trained decision trees and predict future links according to the averaged probability.

6.3.2 Combining Link Prediction and Object Ranking

In this section, we introduce the learning-to-rank method to predict object ranking in a future network. The proposed method uses results of link prediction. We use the following object ranking methods for the detection of promising objects. We defined the combination of ten decision trees as a link predictor.

One Link Predictor (ONE)
ONE uses only a single link predictor to predict future links. After predicting links, we calculate the network centralities and rank objects according to their centralities. This method serves as a baseline.

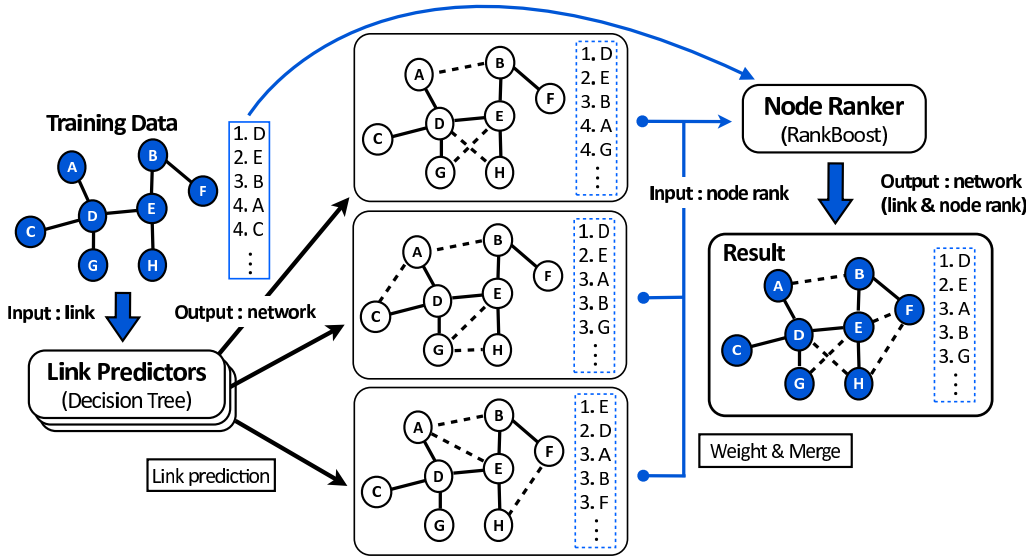

Figure 6.4: Flow chart of how to predict the author rank using RankBoost.

Multiple Link Predictor (MUL)

MUL uses multiple link predictors and predicts multiple future networks. The averaged ranking in predicted networks is a final ranking. However, this method equally weights all link predictors.

RankBoost (RB)

RB re-ranks the results of MUL using RankBoost [24]. RB weights the results of link predictors according to object ranking based on the predicted network centralities. As described in Section 6.2.2, the link prediction with importance of object predicts the future object ranking precisely. Figure 6.4 presents a flowchart of RB.

In summary, ONE uses a single link predictor. MUL uses multiple link predictors and averages them. Although MUL does not weight link predictors, RB weights them with the object ranking based on network centralities. The next section provides a brief overview of RankBoost [24].

6.3.3 RankBoost

RankBoost is a learning-to-rank algorithm that is widely used for re-ranking in various fields, for example information retrieval [45], tag recommendation [79], and facial expression recognition [80]. RankBoost trains one weak ranker at each

round of iteration minimizing the loss function in Equation (6.1), and combines these weak rankers as the final ranking function. After each round, the object pairs are re-weighted, it decreases the weight of correctly ranked pairs and increases the weight of incorrectly ranked pairs.

$$rloss_D(H) = \sum_{x_0,x_1} D(x_0,x_1)\delta(H(x_1) \leq H(x_0)), \tag{6.1}$$

$$D(x_0,x_1) = max(0,\Phi(x_0,x_1)),$$

where H is a linear sum of weak learners, called a strong learner, and $\delta(\pi)$ is the function which returns 1 if π is true and 0 otherwise, and $\Phi(\cdot)$ is a feedback function defined as $\Phi : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, and if x_1 is higher than x_0, $\Phi(x_0,x_1) > 0$. The upper limit of the rank loss function Z_t is also defined as

$$Z_k = \sum_{x_0,x_1} D_k(x_0,x_1)\exp(\alpha_k(h_k(x_0) - h_k(x_1))),$$

where h_k is the output of k^{th} weak learner, corresponding to author rank derived from link prediction, and α_k is the weight of weak learner h_k, which are selected to minimize Z_k.

6.3.4 Weighting Link Predictors

This section presents a description of how RankBoost weights link predictors and their object ranking. When a network g^t at time t is given, our goal is to obtain the object ranking function $H(o)$ made from K future networks $g^{t+\Delta t}$ at time $t + \Delta t$. These networks are predicted by K link predictors. RankBoost weights the K link predictors according to object ranking derived from each predictor. The weight assignment process is the following. First, we predict a network $g_k^{t+\Delta t}$ using k-th link predictor. We estimate the object ranking $h_k(o)^{t+\Delta t}$ on each network. Second, the distribution is initialized D by feedback function $\Phi(\cdot)$. In this case, we define a ranking function of objects based on network centralities as $rank(\cdot)$, and also define a feedback function as $\Phi(x_0,x_1) = rank(x_1) - rank(x_0)$. The feedback function heavily weights object pairs for which the rankings differ considerably. The weight of each link predictor is estimated as follows. We choose a pair of objects (o_i, o_j) from distribution D, and then predict L_k. Each predicted network generates object ranking h_k. Third, the sum r of degree of coincidence weighted D_k is obtained. Then the weight α_k of k's link predictor is determined by r. The distribution D is updated with α_k and the order of object pairs is predicted. Finally, each output h_k of the k-th link predictor is weighted and merged with $\alpha_1,\alpha_2,\ldots,\alpha_K$ of K's instances. Therefore, we obtain the result of the final object rank. **Algorithm1** portrays the Rankboost pseudo-code.

Algorithm1 Rankboost training process

Input: Given entities $e_1, \cdots, e_m \in \mathcal{E}$,
distribution D over $\mathcal{E} \times \mathcal{E}$ and a graph g^t at time t
where
$\mathcal{E}$ is the set of the entities
Initialize $D_1 = D$
Generating link predictors $\{L\}_K$ from the graph g^t
where
$\{L\}_K$ is the function giving link occurrence probabilities to all entity pairs
Predicting K-th graph g^{t+1} from $\{L\}_K$
for $k = 1, \cdots, K$ **do**
- Select pair $(e_i, e_j) \in \mathcal{E} \times \mathcal{E}$ with distribution D
- Get weak ranking h_k from the graph g_k^{t+1}
- Update: $\alpha_k = \frac{1}{2}\ln(\frac{1+r}{1-r})$,
  where $r = \sum_{e_i,e_j} D_k(h_k(e_i) - h_k(e_j))$
- Update: $D_{k+1}(e_i, e_j) = \frac{D_k(e_i,e_j)\exp(\alpha_k(h_k(e_i)-h_k(e_j)))}{Z_k}$
  where Z_k is a normalization factor.
  $Z_k = \sum_{e_i,e_j} D_k(e_i, e_j)\exp(\alpha_k(h_k(e_i) - h_k(e_j)))$

end for
Output the final ranking: $H(e) = \sum_{k=1}^{K} \alpha_k h_k(t)$ and
a strong predictor $\mathbf{L} = \sum_{k=1}^{K} \alpha_k L_k$

6.4 Evaluation

6.4.1 Experimental Setup

We evaluate our proposed methods using arXiv(hep-th) [75] dataset and predict future authorships and author ranking. For object ranking, we use 2,950 authors who had at least one collaborator in 1994. The link indexes described in Section 6.3.1 are extracted from co-authorship network during 1994–1998. The class labels are also extracted from co-authorships in 1999. Link predictors use link indexes extracted during 1995–1999 as features and use true links during 2000–2003 as class labels. RB weights link predictors with importance of objects in 2000.

Table 6.1: AUC by link prediction methods.

JAC	COM	ADA	PRE	KAT	ONE	MUL	RB
0.6561	0.6564	0.6564	0.8435	0.8475	0.7646	0.8066	0.8069

Table 6.2: Results of object ranking by link prediction methods using the link index.

Top	Centrality	BASE	JAC	COM	ADA	PRE	KAT
5	Betweenness	0.0000	0.0000	0.0000	0.2000	0.2000	0.2000
5	Closeness	0.2000	0.6000	0.6000	**0.8000**	0.7379	**0.8000**
5	Degree	0.4444	0.5893	0.5893	0.6804	0.4444	0.4444
5	PageRank	0.2000	0.4000	0.2000	0.4000	0.2000	0.2000
10	Betweenness	0.2889	0.3778	0.4222	0.4667	0.4667	0.5111
10	Closeness	0.5111	0.6000	0.5111	0.6444	0.6293	0.6444
10	Degree	0.7383	0.7916	0.8236	0.8837	0.7621	0.8098
10	PageRank	0.4667	0.5111	0.6000	0.6444	0.4667	0.5111
20	Betweenness	0.3474	0.3895	0.3789	0.4526	0.4526	0.4421
20	Closeness	0.3747	0.4656	0.4274	0.4697	0.4380	0.4274
20	Degree	0.6093	0.6223	0.6407	0.6486	0.6278	0.6748
20	PageRank	0.6526	0.6947	0.7579	0.7263	0.6526	0.6947

6.4.2 Experimental Results

Link Prediction

We evaluate the accuracy of link prediction using co-authorships that occurred during 2000–2003 as true links, and others as false links. As an evaluation metric, we used the Area Under the ROC Curve (AUC). The value of AUC ranges from 0 to 1. High AUC values mean that link predictors predict true links precisely. In addition, AUC = 0.5 means random link prediction. When AUC = 1, link predictors entirely predict true links. For RB, we tuned the parameter of RankBoost in best performance in this experiment.

Table 6.1 shows the AUC by link prediction methods. In this Table, JAC, COM, ADA, PRE, and KAT respectively represent link prediction methods using the Jaccard coefficient, common neighbor, Adamic/Adar, Preferential Attachment, and $Katz_\beta$ for link prediction. ONE, MUL, and RB respectively use a single-link predictor, multiple link predictors, and weighted multiple link predictors. Table 6.1 shows that $Katz_\beta$ outperformed all link prediction methods in AUC because ONE, MUL, and RB all use link indexes including poor indexes such as JAC, COM, and ADA. Then, we compare the link predictors: ONE, MUL, and RB. Results show

Table 6.3: Results of object ranking using the proposed methods.

Top	Centrality	ONE	MUL	RB
5	Betweenness	0.0067	0.0000	**0.9933***
5	Closeness	0.5133	**0.8000**	**0.8000**
5	Degree	0.5404	0.7379	**0.9487***
5	PageRank	0.3133	0.4000	**0.8200***
10	Betweenness	0.3956	0.4222	**0.6326***
10	Closeness	0.5111	0.5556	**0.6919***
10	Degree	0.7944	0.8866	**0.9290***
10	PageRank	0.5541	0.6889	**0.8000***
20	Betweenness	0.3828	0.4000	**0.5382***
20	Closeness	0.3811	0.4063	**0.5439***
20	Degree	0.6225	0.6319	**0.6815***
20	PageRank	0.6712	0.7368	**0.7632***

that the AUC of bagging of 30 link predictors (MUL) outperformed one link predictor (ONE) in AUC by 8 %. The difference is statistically significant as assessed using a t-test ($p < 0.05$). This result suggests that using multiple link predictors improves the link prediction performance. Additionally, we compared MUL to RB, which weights the results of MUL using RankBoost. Although MUL and RB use the same link predictors, the AUC of RB is slightly higher than that of MUL. Nevertheless, no statistical significance exists between MUL and RB.

Object Ranking

In this section, we evaluate the performance of object ranking. Tables 6.2 and 6.3 show Kendall's rank correlation coefficients between the predicted object importance and true ones. We use top 5, 10, and 20 objects ranked by their network centralities. The best performance run is shown in bold, with significant improvement as assessed using the Wilcoxon test (p <0.05). It is marked with * over ONE.

When comparing the results of link prediction methods using the link index to that of BASE, which uses no link prediction and object ranking methods, all methods outperformed BASE in the correlation coefficient for all network centralities. The results suggest that predicting links using the link index is effective to predict the future object ranking.

We discuss the relation between the network centrality and the link index. In general, the Degree and PageRank values become high when many links connect to

the object. Consequently, for predicting the importance of these centralities, it is necessary to predict the links that will be connected to popular objects. In contrast, Closeness and Betweenness values become highest when a target object lies in the shortest path of pairs of objects. Therefore, for predicting the importance of these centralities, it is necessary to predict the shortest path between objects. All link prediction methods use link indexes described in Section 6.3.1. These indexes depend on the link structure around objects, so that the number of links connected to objects affects the value of the link index. Results show that link prediction methods using common neighbors (COM), Jaccard coefficient (JAC), Adamic/Adar (ADA), preferential attachment (PRE), and $Katz_\beta$ (KAT) can predict an object's importance in a future network.

We compare three methods using link predictors. RB, MUL, and ONE use the same features. However, their prediction of future object ranking methods differs. Table 6.3 shows that MUL outperformed ONE in all network centralities because MUL also outperformed ONE in AUC by link prediction described in Table 6.1. Additionally, we found that RB outperformed MUL in all cases even though RB and MUL use the same link predictors. However, RB uses importance objects for object ranking which MUL does not. The results suggest that using the importance of objects is effective to predict the importance of objects in a future network.

6.5 Summary

In this chapter, we proposed the object ranking method using link prediction for a dataset, where its network structure changes dynamically over time. Our findings are the following. First, we found that adding true links to an original network engenders prediction of precise object importance in a future network. Second, we found that adding true links that connect to important objects has the power to predict promising objects. Based on these findings, we proposed an object ranking method that combines link prediction and the learning-to-rank algorithm. The experimentally obtained results obtained using arXiv (hep-th) datasets demonstrate that the proposed method outperformed link prediction methods using standard link indexes to predict the importance of objects in a future network.

The next chapter of this dissertation presents the findings of this dissertation and proposes some potential directions for future work.

Chapter 7

Conclusion and Future Work

In this chapter, we conclude this dissertation and provide a broad perspective on our work. In Section 7.1, we review the process of time-aware information retrieval. Then, in Section 7.2 we summarize the experimentally obtained results reported in this dissertation. We conclude the chapter and the dissertation in Section 7.3, where we discuss potential directions for future research.

7.1 Overview of Time-Aware Information Retrieval in Social Networks

This dissertation describes time-aware information retrieval in social networks, which includes word-based temporal relevance model concept-based temporal relevance model, interactive temporal IR, and time-aware object ranking. To summarize the proposed methods described in this dissertation, we review these steps in this section.

- In Chapter 3, we describe word-based temporal relevance feedback for query expansion. In social networks, time plays an important role in finding topic-related terms and relevant documents because many topics are described actively at a specified time. Our proposed method combined two temporal properties (temporal variation and recency) to retrieve topic-related time-stamped documents.
- In Chapter 4, we described a concept-based query expansion method based on a temporal relevance model that uses the temporal variation of multi-term concepts. Unlike existing retrieval models that use only lexical information

of words or concepts, the proposed model combines lexical and temporal properties effectively by modeling temporal variations of concepts in social networks.

- In Chapter 5, we described two-stage relevance feedback approaches to overcome the limitation of pseudo-relevance feedback. First, the user selects only one relevant document among top ranked initial search results and combines it with an original user query for relevance feedback, where the combined query is used for re-retrieving documents. Second, to improve search results further, the pseudo-relevance feedback using a document dependent temporal relevance model is applied to top ranked re-retrieved documents.

- In Chapter 6, we described a framework to predict the future significance or importance of users in social networks through link prediction. The proposed method first predicts future links between nodes by multiple supervised classifiers and applies the RankBoost algorithm for combining predictions such that the links would engender more precise predictions of a centrality (significance) measure of our choice.

7.2 Summary of Experimental Results

In this section, we summarize some key experimentally obtained results presented in this dissertation. Because the focus of this dissertation is on queries issued by users in social networks, we mainly report the results of our experiments related to microblog queries in this section.

- Our experimentally obtained results in Chapter 3 using the Tweets2011 corpus and TREC 2011 microblog track queries demonstrated that the proposed word-based query expansion method which combines two time-aware methods efficiently improves the retrieval performance in Average Precision (AP) by 10.3% for *allrel* relative to an initial search.

- Our experimentally obtained results from Chapter 4, obtained using the Tweets2011 corpus and TREC 2011 and 2012 microblog track queries demonstrate that the concept-based query expansion method based on temporal relevance model significantly improves retrieval performance in AP by 24.4% for *allrel* and 30.7% for *highrel* compared to the initial search.

- Our experimentally obtained results in Chapter 5 using the Tweets2011 corpus and TREC 2011 and 2012 microblog track queries demonstrated that

two-stage relevance feedback considerably improved retrieval performance in AP by 49.9% for *allrel* and 33.6% for *highrel* compared to the initial search.

- Our experimentally obtained results presented in Chapter 6 using a co-authorship network extracted from the arXiv (hep-th) [75] dataset demonstrated that the proposed time-aware object-ranking method using RankBoost can precisely estimate both future links and the future ranking of objects comparing to standard link-prediction methods.

7.3 Future Work

Time-aware information retrieval that incorporates temporal properties into its models will become increasingly important in the field of IR in the future because social network services are rapidly generating huge amounts of time-stamped documents. However, retrieval for such time-stamped documents presents many difficult research challenges that are not addressed in this dissertation. Next, we describe some of these challenges and directions for potential future research.

Context Modeling for IR In this dissertation, we described word and concept based query expansion methods using temporal pseudo-relevance feedback (PRF). Although our temporal PRF methods are effective for retrieving time-stamped documents, they sometimes failed to outperform the PRF methods that used only lexical information when pseudo-relevant documents failed to indicate topically relevant time. To address this problem, we plan to develop a temporal PRF method by tracking the context of a given topic instead of tracking the word or concept over time. We hypothesize that context modeling such as Latent Semantic Indexing [17], probabilistic Latent Semantic Indexing [25], Latent Dirichlet Allocation [10], tracking [76], and deep learning [26] can overcome the ambiguity of words and concepts, so that modeling the context estimates the precision of the topically relevant time and further improves the retrieval performance.

Temporal Summarization This dissertation emphasizes ad-hoc search. However, it is important to examine *temporal summarization* [3] that specifically incorporates timeliness and redundancy simultaneously. For temporal summarization, the corpus of time-stamped documents will be regarded as a stream and IR systems sequentially summarize stream data related to unexpected news events such as earthquakes and natural disasters. Similar to our

work in this dissertation, the temporal summarization task requires modeling of a real-time nature on the Web for classifying time-stamped data as relevant or not. Consequently, our time-aware information retrieval techniques can be important for developing temporal summarization.

Bibliography

[1] L. Adamic and E. Adar. Friends and neighbors on the web. *Social networks*, 25(3):211–230, 2003.

[2] G. Amati, G. Amodeo, and C. Gaibisso. Survival analysis for freshness in microblogging search. In *Proceedings of the 21st ACM international conference on information and knowledge management*, pages 2483–2486, 2012.

[3] J. Aslam, F. Diaz, M. Ekstrand-Abueg, V. Pavlu, and T. Sakai. TREC 2013 temporal summarization. In *Proceedings of the 22nd text retrieval conference*, 2013.

[4] A. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509, 1999.

[5] M. Bendersky and W. B. Croft. Discovering key concepts in verbose queries. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval*, pages 491–498, 2008.

[6] M. Bendersky and W. B. Croft. Modeling higher-order term dependencies in information retrieval using query hypergraphs. In *Proceedings of the 35th annual international ACM SIGIR conference on research and development in information retrieval*, pages 941–950, 2012.

[7] M. Bendersky, D. Metzler, and W. B. Croft. Learning concept importance using a weighted dependence model. In *Proceedings of the 3rd ACM international conference on Web search and data mining*, pages 31–40, 2010.

[8] M. Bendersky, D. Metzler, and W. B. Croft. Parameterized concept weighting in verbose queries. In *Proceedings of the 34th annual international ACM SIGIR conference on research and development in information retrieval*, pages 605–614, 2011.

[9] M. Bendersky, D. Metzler, and W. B. Croft. Effective query formulation with multiple information sources. In *Proceedings of the 5th ACM international conference on Web search and data mining*, pages 443–452, 2012.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine learning research*, 3:993–1022, Mar. 2003.

[11] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine* 1. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[12] C. Buckley and E. M. Voorhees. Evaluating evaluation measure stability. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval*, pages 33–40, 2000.

[13] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval*, pages 25–32, 2004.

[14] G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval*, pages 243–250, 2008.

[15] J. Choi and W. B. Croft. Temporal models for microblogs. In *Proceedings of the 21st ACM international conference on information and knowledge management*, pages 2491–2494, 2012.

[16] W. Dakka, L. Gravano, and P. G. Ipeirotis. Answering general time-sensitive queries. *IEEE transactions on knowledge and data engineering*, 24(2):220–235, 2012.

[17] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[18] F. Diaz. Integration of news content into web results. In *Proceedings of the 2nd ACM international conference on Web search and data mining*, pages 182–191, 2009.

[19] M. Efron. Query-specific recency ranking: survival analysis for improved microblog retrieval. In *Proceedings of SIGIR 2012 workshop on time-aware information access*, 2012.

[20] M. Efron and G. Golovchinsky. Estimation methods for ranking recent information. In *Proceedings of the 34th annual international ACM SIGIR conference on research and development in information retrieval*, pages 495–504, 2011.

[21] M. Efron, A. Kehoe, P. Organisciak, and S. Suh. The university of illinois' graduate school of library and information science at TREC 2011. In *Proceedings of the 20th text retrieval conference*, 2011.

[22] M. Efron, P. Organisciak, and K. Fenlon. Improving retrieval of short texts through document expansion. In *Proceedings of the 35th annual international ACM SIGIR conference on research and development in information retrieval*, pages 911–920, 2012.

[23] L. Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1979.

[24] Y. Freund, R. Iyer, R. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *The journal of machine learning research*, 4:933–969, 2003.

[25] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval*, pages 50–57, 1999.

[26] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on information and knowledge management*, pages 2333–2338, 2013.

[27] N. Japkowicz et al. Learning from imbalanced data sets: a comparison of various strategies. In *AAAI workshop on learning from imbalanced data sets*, volume 68, 2000.

[28] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM transactions on information systems*, 20(4):422–446, 2002.

[29] R. Jones and F. Diaz. Temporal profiles of queries. *ACM transactions on information systems*, 25(3), 2007.

[30] I. G. Kalmanovich and O. Kurland. Cluster-based query expansion. In *Proceedings of the 32nd annual international ACM SIGIR conference on research and development in information retrieval*, pages 646–647, 2009.

[31] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[32] M. Keikha, S. Gerani, and F. Crestani. Time-based relevance models. In *Proceedings of the 34th annual international ACM SIGIR conference on research and development in information retrieval*, pages 1087–1088, 2011.

[33] S. Kitaguchi, T. Miyanishi, K. Seki, and K. Uehara. Interactive disaster information search system for microblog by minimal user feedback. In *Proceedings of the 9th asia information retrieval societies conference*, pages 476–487, 2013.

[34] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[35] W. R. Knight. A computer method for calculating kendall's tau with ungrouped data. *Journal of the american statistical association*, 61(314):436–439, 1966.

[36] R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval*, pages 191–202, 1993.

[37] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval*, pages 194–201, 2004.

[38] O. Kurland, L. Lee, and C. Domshlak. Better than the real thing? iterative pseudo-query processing using cluster-based language models. In *Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval*, pages 19–26, 2005.

[39] H. Lang, D. Metzler, B. Wang, and J.-T. Li. Improved latent concept expansion using hierarchical Markov random fields. In *Proceedings of the 19th ACM international conference on information and knowledge management*, pages 249–258, 2010.

[40] V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval*, pages 120–127, 2001.

[41] M. Lease. An improved Markov random field model for supporting verbose queries. In *Proceedings of the 32nd annual international ACM SIGIR conference on research and development in information retrieval*, pages 476–483, 2009.

[42] X. Li and W. Croft. Time-based language models. In *Proceedings of the 12th international conference on information and knowledge management*, pages 469–475, 2003.

[43] F. Liang, R. Qiang, and J. Yang. Exploiting real-time information retrieval in the microblogosphere. In *Proceedings of the 12th ACM/IEEE-CS joint conference on digital libraries*, pages 267–276, 2012.

[44] J. Lin and M. Efron. Temporal relevance profiles for tweet search. In *Proceedings of SIGIR 2013 workshop on time-aware information access*, 2013.

[45] T. Liu, J. Xu, T. Qin, W. Xiong, and H. Li. Letor: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of SIGIR 2007 workshop on learning to rank for information retrieval*, pages 3–10, 2007.

[46] X. Liu, J. Bollen, M. Nelson, and H. Van de Sompel. Co-authorship networks in the digital library research community. *Information Processing & Management*, 41(6):1462–1480, 2005.

[47] X. Liu and W. B. Croft. Cluster-based retrieval using language models. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval*, pages 186–193, 2004.

[48] Y. Lv and C. Zhai. Positional relevance model for pseudo-relevance feedback. In *Proceedings of the 33rd annual international ACM SIGIR conference on research and development in information retrieval*, pages 579–586, 2010.

[49] C. Manning and H. Schütze. *Foundations of statistical natural language processing*, volume 59. MIT Press, 1999.

[50] K. Massoudi, M. Tsagkias, M. de Rijke, and W. Weerkamp. Incorporating query expansion and quality indicators in searching microblog posts. In *Proceedings of the 33th annual european conference on information retrieval*, pages 362–367, 2011.

[51] D. Metzler and C. Cai. USC/ISI at TREC 2011: Microblog track. In *Proceedings of the 20th text retrieval conference*, 2011.

[52] D. Metzler, C. Cai, and E. Hovy. Structured event retrieval over microblog archives. In *Proceedings of the 12th annual conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 646–655, 2012.

[53] D. Metzler and W. B. Croft. Combining the language model and inference network approaches to retrieval. *Information processing & management*, 40(5):735–750, 2004.

[54] D. Metzler and W. B. Croft. A Markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval*, pages 472–479, 2005.

[55] D. Metzler and W. B. Croft. Latent concept expansion using Markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval*, pages 311–318, 2007.

[56] M. Mitra, A. Singhal, and C. Buckley. Improving automatic query expansion. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval*, pages 206–214, 1998.

[57] T. Miyanishi, K. Seki, and K. Uehara. Combining recency and topic-dependent temporal variation for microblog search. In *Proceedings of the 35th annual european conference on information retrieval*, pages 331–343, 2013.

[58] T. Miyanishi, K. Seki, and K. Uehara. Improving pseudo-relevance feedback via tweet selection. In *Proceedings of the 22nd ACM international conference on information and knowledge management*, pages 439–448, 2013.

[59] M. Newman. Clustering and preferential attachment in growing networks. *Physical review E*, 64(2):025102, 2001.

[60] J. O'Madadhain, J. Hutchins, and P. Smyth. Prediction and ranking algorithms for event-based network data. *ACM SIGKDD explorations newsletter*, 7(2):23–30, 2005.

[61] I. Ounis, C. Macdonald, J. Lin, and I. Soboroff. Overview of the TREC-2011 microblog track. In *Proceedings of the 20th text retrieval conference*, 2011.

[62] M.-H. Peetz and M. de Rijke. Cognitive temporal document priors. In *Proceedings of the 35th annual european conference on information retrieval*, pages 318–330, 2013.

[63] M. H. Peetz, E. Meij, M. de Rijke, and W. Weerkamp. Adaptive temporal query modeling. In *Proceedings of the 34th annual european conference on information retrieval*, pages 455–458, 2012.

[64] J. Ponte and W. Croft. 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, 1998.

[65] J. Quinlan. *C4. 5: programs for machine learning*. Morgan Kaufmann, 1993.

[66] S. E. Robertson and K. Sparck Jones. Document retrieval systems. chapter Relevance Weighting of Search Terms, pages 143–160. 1988.

[67] S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on research and development in information retrieval*, pages 232–241, 1994.

[68] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System*, pages 313–323, 1971.

[69] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th international conference on World Wide Web*, pages 851–860, 2010.

[70] H. Sayyadi and L. Getoor. FutureRank: ranking scientific articles by predicting their future PageRank. In *Proceedings of the 9th SIAM international conference on data mining*, pages 533–544. Citeseer, 2009.

[71] M. D. Smucker, J. Allan, and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM international conference on information and knowledge management*, pages 623–632, 2007.

[72] I. Soboroff, I. Ounis, and J. Lin. Overview of the TREC-2012 microblog track. In *Proceedings of the 21st text retrieval conference*, 2012.

[73] T. Strohman, D. Metzler, H. Turtle, and W. Croft. Indri: a language model-based search engine for complex queries. In *Proceedings of the 1st international conference on intelligent analysis*, pages 2–6, 2005.

[74] T. Tao, X. Wang, Q. Mei, and C. Zhai. Language model information retrieval with document expansion. In *Proceedings of the 6th annual conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 407–414, 2006.

[75] The Knowledge Discovery Laboratory, University of Massachusetts Amherst. Dataset: Hep-th. http://kdl.cs.umass.edu/data/hepth/hepth-info.html.

[76] X. Wang and A. McCallum. 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, 2006.

[77] X. Wei and W. B. Croft. 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*, pages 178–185, 2006.

[78] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 261–270, 2010.

[79] L. Wu, L. Yang, N. Yu, and X. Hua. Learning to tag. In *Proceedings of the 18th international conference on World Wide Web*, pages 361–370, 2009.

[80] P. Yang, Q. Liu, and D. Metaxas. Rankboost with l1 regularization for facial expression recognition and intensity estimation. In *Proceedings of the IEEE 12th international conference on computer vision*, pages 1018–1025, 2009.

[81] C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the 10th ACM international conference on information and knowledge management*, pages 403–410, 2001.

[82] C. Zhai and J. Lafferty. Two-stage language models for information retrieval. In *Proceedings of the 25th Annual International ACM SIGIR conference on research and development in information retrieval*, pages 49–56, 2002.

[83] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM transactions on information systems*, 22(2):179–214, 2004.

[84] J. Zhang, M. Ackerman, and L. Adamic. Expertise networks in online communities: structure and algorithms. In *Proceedings of the 16th international conference on World Wide Web*, pages 221–230. ACM, 2007.

Publication list

International conference

[1] Taiki Miyanishi, Sayaka Kitaguchi, Kazuhiro Seki, and Kuniaki Uehara. TREC 2013 Microblog Track Experiments at Kobe University. In *Proceedings of the 21st text retrieval conference*, 2013.

[2] Sayaka Kitaguchi, Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. Interactive Disaster Information Search System for Microblog by Minimal User Feedback. In *Proceedings of the 9th asia information retrieval societies conference*. pp. 476–487, 2013.

[3] Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. Improving Pseudo-Relevance Feedback via Tweet Selection. In *Proceedings of the 22nd ACM international conference on information and knowledge management*. pp. 439–448, 2013.

[4] Taiki Miyanishi and Tetsuya Sakai. Time-aware Structured Query Suggestion. In *Proceedings of the 35th international ACM SIGIR conference on research and development in information retrieval*. pp. 809–812, 2013. (short paper)

[5] Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. Combining Recency and Topic-Dependent Temporal Variation for Microblog Search. In *Proceedings of the 35th european conference on information retrieval*. pp. 331–343, 2013.

[6] Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. TREC 2012 Microblog Track Experiments at Kobe University. In *Proceedings of the 21st text retrieval conference*, 2012. *Fourth place among 33 groups in the Real-time adhoc task.*

[7] Taiki Miyanishi, Naoto Okamura, Xiaoxi Liu, Kazuhiro Seki, and Kuniaki Uehara. TREC 2011 Microblog Track Experiments at Kobe University. In *Proceedings of the 20th text retrieval conference*, 2011. *Fourth place among 58 groups in the Realtime adhoc task. Selected as a speaker for the plenary session*.

[8] Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. Hypothesis Generation and Ranking Based on Event Similarities. In *Proceedings of the 25th annual ACM symposium on applied computing*, pp. 1552–1558, 2010.

Domestic journal

[9] 宮西大樹，関和広，上原邦昭．マイクロブログ文書の選択による適合フィードバックを用いた疑似適合フィードバックの検索性能改善，情報処理学会論文誌，55 巻，5 号，2014．

[10] 宮西 大樹，関和広，上原邦昭．マイクロブログ検索のための時間情報と非時間情報を統合したクエリ拡張，情報処理学会論文誌，54 巻，4 号，pp. 1655-1666, 2013 年 4 月．

[11] 宮西 大樹，関和広，上原邦昭．リンク予測を基にした時系列ネットワーク中でのオブジェクトランキング，人工知能学会論文誌，27 巻，3 号，pp. 223-234．2012 年 3 月．

[12] Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara. Hypothesis Ranking Based on Semantic Event Similarities. *IPSJ transactions on bioinformatics*. Vol.4, pp.9-20, May 2011.

Domestic conference

[13] 宮西大樹，関和広，上原邦昭．コンセプト追跡を用いたマイクロブログ検索．第 6 回 Web とデータベースに関するフォーラム (WebDB Forum 2013)，2013 年 11 月．優秀論文賞，学生奨励賞受賞

[14] 宮西大樹，関和広，上原邦昭．マイクロブログ文書の選択による擬似適合フィードバック．第 157 回 DBS・第 111 回 IFAT 合同研究発表会，2013 年 7 月．学生奨励賞受賞

[15] 北口沙也香，宮西大樹，関和広，上原邦昭．マイクロブログ文書の選択による対話的な災害情報検索システム．言語処理学会第 19 回年次大会，2013 年 3 月．

[16] 北口沙也香，熊南昂司，藤川和樹，宮西大樹．Twitter を利用した震災情報検索支援システム．東日本大震災ビッグデータワークショップ - Project 311 - ，2012 年 10 月．

[17] 宮西大樹，関和広，上原邦昭．生物医学要素関係間の意味的類似度に基づく仮説の順位付け．情報処理学会第 26 回バイオ情報学研究会，No.1，2011 年 9 月．

[18] 宮西大樹，関和広，上原邦昭．ネットワーク構造解析に基づく有望ノードの予測．情報処理学会第 82 回数理モデル化と問題解決研究会，No. 3, 2011 年 3 月．

[19] 宮西大樹，関和広，上原邦昭．リンク予測に基づく有望ノードの同定．第 24 回人工知能学会全国大会， 2010 年 6 月．

[20] 宮西大樹，関和広，上原邦昭．イベント類似度にもとづく仮説の順位付け．平成 22 年度情報処理学会関西支部支部大会，2009 年 8 月．

[21] 宮西大樹，関和広，上原邦昭．生物医学文献からの知識抽出とイベント間のつながりを考慮した発見性を伴う仮説の提示．第 1 回データ工学と情報マネジメントに関するフォーラム／第 7 回日本データベース学会年次大会，2009 年 3 月．