Combining Recency and Topic-Dependent Temporal Variation for Microblog Search

Taiki Miyanishi, Kazuhiro Seki, and Kuniaki Uehara

Graduate School of System Informatics Kobe University, Japan
{miyanishi,seki,uehara}@ai.cs.kobe-u.ac.jp

Abstract. The appearance of microblogging services has led to many short documents being issued by crowds of people. To retrieve useful information from among such a huge quantity of messages, query expansion (QE) is usually used to enrich a user query. Some QE methods for microblog search utilize 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 recently occurred. Our approach leverages temporal properties for QE and combines them according to the temporal variation of a given topic. Experimental results show that this QE method using automatically combined temporal properties is effective at improving retrieval performance.

1 Introduction

Microblogging is one of the most powerful online media for enabling people to understand what is happening around the world today. Among different microblogging services, Twitter [1] is a well-known online social One of the interesting properties of Twitter is that many tweets (messages issued by Twitter users) are posted by crowds of people when a notable event occurs. As a result, a set of tweets about the topic is an important clue about what topics are being actively mentioned at a particular time. For example, when the news that "BBC World Service planned to close five of its language services" [2] was reported from January 25 to 27, 2011, many tweets about this event were actively posted at around this period. To clarify this temporal property of microblogging, we took four topics used in the TREC 2011 Microblog track [10]: "*BBC World Service staff cuts*" (MB001), "*White Stripes breakup*" (MB017), "*Emanuel residency court rulings*" (MB021), and "*US unemployment*" (MB026). Kernel density estimates of the time-stamps of tweets relevant to these four topics are shown in Figure 1. Not all of the temporal variations of a given topic are the same; moreover, many tweets are issued by users during the specified time period. Note that documents relevant to a given topic contain topic-related terms that appear frequently while the topic is being mentioned. For example, the tweets relevant to topic MB001 contain query terms: *BBC*, *cuts*, and *staff* as well as topic-related terms: *axe*

[1] https://twitter.com/
[2] http://www.bbc.co.uk/news/entertainment-arts-12277413

P. Serdyukov et al. (Eds.): ECIR 2013, LNCS 7814, pp. 331–343, 2013.

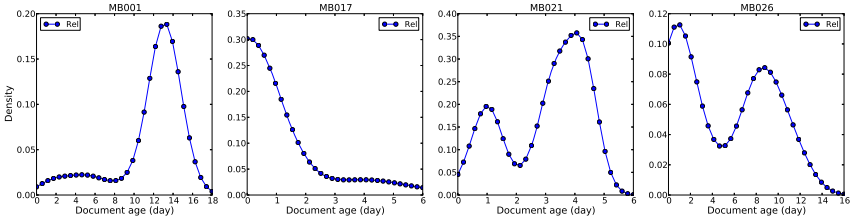

Fig. 1. Temporal variations of four topics (MB001, MB017, MB021, and MB026) from the TREC 2011 Microblog track based on relevant tweets. The x-axis shows document age from query time to document time-stamp. The y-axis shows the kernel-estimated probability density for the document age. A high density indicates the period in which the topic was actively mentioned.

and *jobs*. The point is that if we could identify when a topic is being actively mentioned, we could also easily detect its relevant documents and related terms.

Besides temporal variation, recency is also an importance temporal property. Some research has incorporated recency into microblog retrieval methods in order to search for recent and relevant tweets posted at around the time a query was issued [3,8]. For example, the method considering recency is effective for retrieving tweets relevant to topics MB017 and MB026 in Figure 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 [2,6]. 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 handle specific temporal variations consisting of an old peak far from the query time or a multimodal temporal variation (e.g., MB001 and MB021 in Figure 1) and consequently cannot discover terms temporally related to these topics. Other language model approaches incorporating temporal variation also performed well [1,7], but they 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. We used the Tweets2011 corpus[3], which consists of more than 16 million tweets over a period of two weeks to verify the effectiveness of our method.

2 Previous Time-Based Microblog Search Methods

Microblog users often search for documents regarding a recent topic concerning an event that happened recently. Documents relevant to some recent topics tend

[3] `http://trec.nist.gov/data/tweets/`

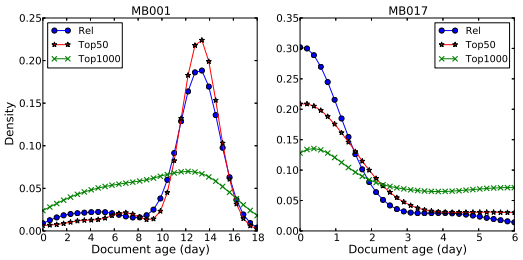

Fig. 2. Two kernel density estimates corresponding to topics MB001 and MB017. The blue line (*Rel*) is the estimate for relevant documents. The red line (*Top50*) and green line (*Top1000*) are the estimates for the top 50 and 1000 retrieved documents, respectively.

to be issued at around the query time (e.g., MB017 and MB026 in Figure 1). Taking advantage of this characteristic, Efron [3] incorporated temporal properties, such as recency and the smoothed temporal variation of a topic, into microblog search. His method used a temporal profile [4] 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 2 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:

$$s(D, Q) = \log P(Q|D) + \phi(T_Q, T_D), \tag{1}$$

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) since it assumes that time-stamps are generated from a Gaussian distribution. Our model for handling any temporal variations and discovering terms

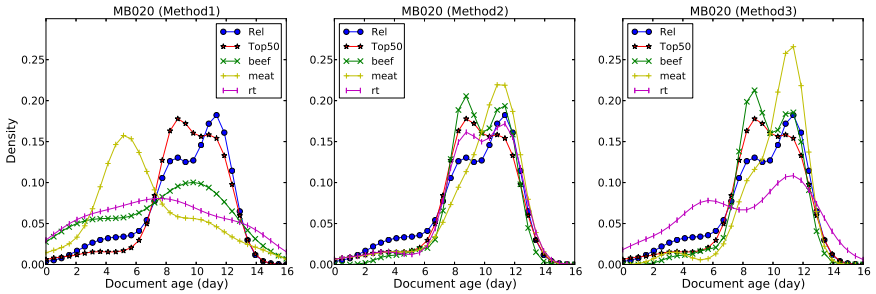

Fig. 3. Three types of kernel density estimates obtained using topic MB020 (Taco Bell filling lawsuit). Green, yellow, and purple lines show the temporal profiles for *beef*, *meat*, and *rt*, respectively. *Top50* and *Rel* are temporal profiles created from the top 50 documents and relevant documents for the topic.

temporally related to a topic for QE is explained in Section 3. It ingeniously combines two types of time-aware QE methods according to the temporal variation of a given topic.

3 Our Approach

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 Section 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[4] from the tweets.

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 2 shows that the query profile (*Top50*) can be regarded as an approximation of the relevant profile (*Rel*). Our assumption is that we can

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

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 with Dirichlet smoothing [13] (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[5]. 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. The left plot (**Method1**) shows that the temporal profile 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 [13] 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.

[5] `http://gizmodo.com/5742413/`

To model the temporal properties of a candidate term combined with a seed query, we borrow Jones and Diaz's idea [4]. 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 $P(t|Q) = \lambda P'(t|Q) + (1 - \lambda)P(t|C)$, 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 [4] 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.

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 (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.

$$S_{TVQE}(w, Q) = -D_{KL}(P(t|w \cap^{+} Q), P(t|Q)) = -\sum_{t=1}^{T} P(t|w \cap^{+} Q) \log \frac{P(t|w \cap^{+} Q)}{P(t|Q)}, \tag{2}$$

where $w \cap^{+} Q$ is the expanded query (produced by **Method3** in Section 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.

To incorporate recency into a QE method, we also use another QE method (TRQE), which is a modification of Efron's model (see Equation (1)) as follows:

$$S_{TRQE}(w, Q) = \phi(T_Q, T_{Q'}) = \log\Big(\frac{m_{T_Q}}{m_{T_{Q'}}}\Big), \tag{3}$$

where $m_{TQ'}$ is the sample mean of the time-stamps 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.

3.2 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), but they only partially consider when a topic is actively mentioned (e.g., MB001 and MB021 in Figure 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.

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 2, 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:

$$\zeta = P(X \leq \gamma) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\gamma} \exp\left\{ \frac{(x-\mu)^2}{2\sigma^2} \right\} dx, \tag{4}$$

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 2) 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:

$$S_{TVRQE}(w, Q) = (1 - \zeta) \cdot S'_{TVQE}(w, Q) + \zeta \cdot S'_{TRQE}(w, Q) \tag{5}$$

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.

4 Evaluation

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. This collection consists of about 16 million tweets sampled between January 23rd and February 7th, 2011. 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.

We indexed tweets posted before the specific time associated with each topic by the Indri search engine with the setting described in Section 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[6] 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 may contain topic-related 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[7].

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 [12] 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 and L for QE is discussed in the next section.

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

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

[7] `https://github.com/shuyo/ldig`

Table 1. Retrieval performance of the QE method (we set $K = 10$, $L = 30$, $M = 30$, $\gamma = 5$). Metzler [9] is the best performance for the realtime adhoc task in the TREC 2011 Microblog track. Liang [7] is a state-of-the-art query modeling approach post TREC 2011.

Method	LM	RF [5]	RQE [8]	TVQE	TRQE	TVRQE	Metzler [9]	Liang [7]
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

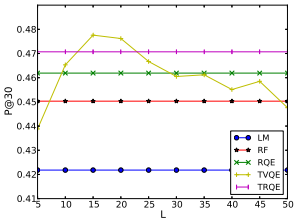

Fig. 4. Length of the temporal profile

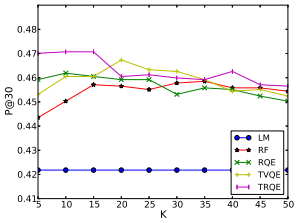

Fig. 5. No. of candidate terms for QE

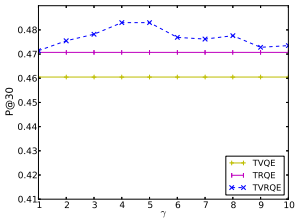

Fig. 6. TVRQE parameter γ

used include precision at rank 30 (P@30) and mean average precision (MAP). P@30 was the official Microblog track metric in 2011 [10]. These measures provided a brief summary of the quality of the retrieved tweets. Note that we used only the top 30 tweets retrieved by each method. To test for statistical significance, we used a 2-tailed paired t-test. The best performing run is indicated in bold; significant improvements are indicated with † and ‡ for $p < 0.05$ against a pseudo-relevance feedback method (RF) [5], which is an Indri's implementation and a recency-based QE method (RQE) for microblog search [8] with the past work's parameters. RF and RQE are topical and temporal QE baselines.

4.2 Experimental 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 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 microblog search performance.

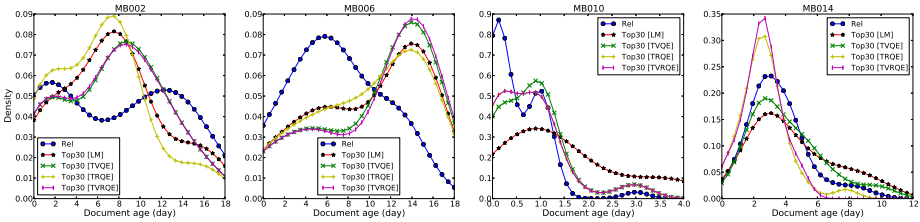

Fig. 7. Kernel density estimates corresponding to four topics: MB002, MB006, MB010, and MB014. The curves, *Rel*, *Top30 [LM]*, *Top30 [TVQE]*, *Top30 [TRQE]*, and *Top30 [TVRQE]*, are estimates for relevant documents; the top 30 documents retrieved by using a seed query and the top 30 re-retrieved documents retrieved by using an expanded query with TVQE, TRQE, and TVRQE.

The degrees of relationship among the parameters (L, K, and γ) of each QE method are shown in Figure 4, 5, and 6. The x-axis shows each parameter. The y-axis shows the values in P@30. Figure 4 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 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 5 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 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 (4), 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.

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 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 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

Table 2. Top 8 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

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]*).

Table 2 lists the top 8 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[8]) that temporal-variation-based method TVQE 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) 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[9], *relationship* in MB006[10], and *anthony* and *thriller* in MB014[11]) 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.

5 Related Work

Microblog search has recently become an attractive research task in the information retrieval (IR) field. Efron et al. [3] showed that the temporal property of

[8] 2022 FIFA World Cup will be held in Qatar.

[9] 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.

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

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

microblogs has the potential to improve retrieval performance. Microblog search requires relevant and most recent documents. Li and Croft [6] incorporated recency into the language model framework for IR. Efron and Golovchinsky [2] proposed IR methods incorporating temporal properties into language modeling and showed their effectiveness for recency queries. Dakka et al. [1] also proposed the general ranking mechanism integrating temporal properties into the language model identifying the important periods. Peetz et al. [11] proposed query modeling using temporal burst, which is similar to our method TVQE. However, Dakka and Peetz's works cannot combine two types of temporal properties (recency and temporal variation) by topic. Our method simultaneously takes account of document freshness and the temporal variation of a topic and can appropriately weight QE methods according to the topic's temporal variation.

Our approach mainly focuses on the QE method because of its simplicity and effectiveness. To refine an original query, Lavrenko's relevance model [5] is commonly used. For microblog search, Massoudi et al. [8] proposed a QE method selecting terms temporally closer to the query time. As far as we know, our QE method is the first that efficiently leverages both recency and temporal variation by topic to discover topic-related terms.

6 Conclusion

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 paper, 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.

References

1. Dakka, W., Gravano, L., Ipeirotis, P.G.: Answering general time-sensitive queries. TKDE 24(2), 220–235 (2012)
2. Efron, M., Golovchinsky, G.: Estimation methods for ranking recent information. In: SIGIR, pp. 495–504 (2011)
3. Efron, M.: The university of illinois' graduate school of library and information science at TREC 2011. In: TREC (2011)
4. Jones, R., Diaz, F.: Temporal profiles of queries. TOIS 25(3) (2007)
5. Lavrenko, V., Croft, W.B.: Relevance based language models. In: SIGIR, pp. 120–127 (2001)

6. Li, X., Croft, W.: Time-based language models. In: CIKM, pp. 469–475 (2003)
7. Liang, F., Qiang, R., Yang, J.: Exploiting real-time information retrieval in the microblogosphere. In: JCDL, pp. 267–276 (2012)
8. Massoudi, K., Tsagkias, M., de Rijke, M., Weerkamp, W.: Incorporating Query Expansion and Quality Indicators in Searching Microblog Posts. In: Clough, P., Foley, C., Gurrin, C., Jones, G.J.F., Kraaij, W., Lee, H., Mudoch, V. (eds.) ECIR 2011. LNCS, vol. 6611, pp. 362–367. Springer, Heidelberg (2011)
9. Metzler, D., Cai, C.: Usc/isi at trec 2011: Microblog track. In: TREC (2011)
10. Ounis, I., Macdonald, C., Lin, J., Soboroff, I.: Overview of the TREC-2011 microblog track. In: TREC (2011)
11. Peetz, M.-H., Meij, E., de Rijke, M., Weerkamp, W.: Adaptive Temporal Query Modeling. In: Baeza-Yates, R., de Vries, A.P., Zaragoza, H., Cambazoglu, B.B., Murdock, V., Lempel, R., Silvestri, F. (eds.) ECIR 2012. LNCS, vol. 7224, pp. 455–458. Springer, Heidelberg (2012)
12. Strohman, T., Metzler, D., Turtle, H., Croft, W.: Indri: a language model-based search engine for complex queries. In: ICIA (2005)
13. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. TOIS 22(2), 179–214 (2004)