World Wide Web (2020) 23:1791–1809
https://doi.org/10.1007/s11280-019-00747-0

Query-based unsupervised learning for improving social media search

Khaled Albishre[1,2] · Yuefeng Li[1] · Yue Xu[1] · Wei Huang[3]

Received: 2 March 2019 / Revised: 31 July 2019 / Accepted: 9 October 2019 /
Published online: 27 November 2019

Abstract
In the current information era over the internet, social media has become one of the essential information sources for users. While the text is the primary information representation, finding relevant information is a challenging mission for researchers due to its nature (e.g., short length, sparseness). Acquiring high-quality search results from massive data, such as social media needs a set of representative query terms that are not always available. In this paper, we propose a novel query-based unsupervised learning model to represent the implicit relationships in the short text from social media. This bridges the gap of the lack of word co-occurrences without requiring many parameters to be estimated and external evidence to be collected. To confirm the proposed model effectiveness, we compare the proposed model with state-of-the-art lexical, topic model and temporal models on the large-scale TREC microblog 2011-2014 collections. The experimental results show that the proposed model significantly improved overall state-of-the-art lexical, topic model and temporal models with the maximum percentage of increase reaching 33.97% based on MAP value and 21.38% based on Precision at top 30 documents. The proposed model can improve the social media search effectiveness in potential closely retrieval tasks, such as question answering and timeline summarisation.

Keywords Information retrieval · Text mining · Microblog retrieval · Pseudo-relevance feedback

1 Introduction

Social media platforms such as Twitter and Facebook have become one of the main information sources among Web applications. The main information representation among these

This article belongs to the Topical Collection: *Computational Social Science as the Ultimate Web Intelligence*
Guest Editors: Xiaohui Tao, Juan D. Velasquez, Jiming Liu, and Ning Zhong

✉ Khaled Albishre
kmbishre@uqu.edu.sa

Extended author information available on the last page of the article.

platforms is the short text. Around 500 million tweets are sent by 335 million active users on the Twitter platform daily. Also, users are seeking new information through social media search engines by submitting an initial query based on their background. In the face of this overwhelming data, it is hard to find sophisticated information that meets users' needs. Therefore, the aim of boosting initial user information needs has identified a challenging research problem to the computational social science field.

Web Intelligence (WI) can be an effective way of distinguishing user needs from weakly evidence with relevant information. A well-known methodology for boosting initial user information needs is Pseudo Relevance Feedback (PRF). The main idea behind PRF is that it uses the top features from the initial ranked documents to expand the original query. One essential component of PRF is the initial ranked documents set. Due to the absence of real judgment (uncertain information in unlabeled data) as to whether the document is relevant or not, it is hard to determine the useful features that can increase the performance of PRF. The quantity of relevant information in the initial retrieved set relies on the original user information needs; if the query is short or vague, more uncertain information can appear the next features selection processes. PRF applied to social media short texts without considering the nature of the texts (e.g., time sensitivity or short length) can introduce more noise features [8, 28, 35]. The short text lacks content due its short length (e.g., 140 characters in a tweet), and there is not enough statistical information to extract its. Since the PRF framework works by selecting a number of the initial ranked documents, very few relevant instances will appear in the initial set. It seems that the short text faces another challenge which is data sparsity.

The need to improve social media search has received much attentions in recent years. As a way to deal with social media's data and time sensitivity, using temporal evidence can enhance search performance [8, 11]. However, relevant features' temporal distributions are not equally uninformed; this could require the need for another extensive task such as bursts or event detection [6]. Due to a lack of content in social media posts, external evidence has been wildly utilised in the literature [26, 35]. Using an external knowledge base increases the time complexity for the learning model. To revise this challenge, some researchers tried to introduce supervised learning to social media search, which required labelled data [37, 40]. Latent Dirichlet Allocation (LDA) topic model [7] is a popular method for deriving high-quality information (latent topics) from text. It is an unsupervised method for a given set of unlabelled documents (e.g., a set of retrieved documents). However, the major process of the LDA algorithm is based on sampling techniques. Therefore, the discovered terms in latent topics are frequent terms. As mentioned in [4, 5], frequent LDA terms are used to understand the focused topics; however, frequent terms are also frequently used in non-relevant documents because retrieved documents include both relevant and non-relevant documents. Thus, it is very difficult to reduce noisy from frequent terms for a query-based unsupervised method.

In term of statistics, unsupervised learning intends to infer prior probability distributions $p(x)$ and supervised learning intends to infer conditional probability distributions $p(x|Y)$ for any input object x based on a large training set Y. Priors can be created using a number of statistical methods (e.g., a normal distribution) or determined from previous experiments. However, in real applications, priors are universal if the relevant background is not taken into account. In this research, we consider the relevant background by using a query (Q); therefore, **unsupervised learning** in this research intends to infer a probability distribution $p(x, Q)$.

We depart from existing methods by observing that the lack of word co-occurrence information in short texts has the main impact on improving the social media search. The ultimate

aim is to capture optimal implicit relationships from the initial retrieved tweets in order to infer more knowledge to serve user needs. As we mentioned before, the critical problem is how to reduce uncertainties in retrieved tweets, because we do not know which tweets are relevant to what the user wants. This paper proposes a new query-based unsupervised method to overcome the limitations of LDA when deriving high-quality terms for retrieved documents. We firstly receive the initial results for a given query and then select the top-ranked tweets. With the top-ranked tweets, we build a new virtual documents space based on query-based tweets pooling strategy to discover a new relationship between the user information needs and the selected tweets. Then, we obtain a novel weight for each word in the selected tweets respect to the implicit relevant evidence. Therefore, we believe that the proposed model will be useful for conducting high-quality unsupervised learning in order to find high-quality text features. We will experimentally prove this assumption in this paper.

The main contributions of this paper[1] include:

1. To deal with data sparsity in initial retrieved tweets, we aggregate tweets based on query-based pooling.
2. To reduce uncertainties in information in the new aggregation technique, we describe the relationships between query and tweets through an intermediate sets (i.e., virtual document space).
3. After discovering the relationships, we estimate the appropriate weight for each term that reflects its discriminatory power.
4. To verify the proposed model performance, we conduct extensive experiments on TREC microblog dataset 2011-2014 comparing the state-of-the-art model and the results outstanding the baseline models for all datasets overall.

To the best of our knowledge, the proposed model is the first research to aggregate tweets based on the appears of query terms and the first to establish the implicit relationships between them in order to improve social media search.

The outline of the paper is as follows. Section 3 shows the problem formulation. Section 4 formally describes the proposed model in detail. Corresponding experimental results are shown in Section 5. Finally, Section 6 concludes the paper.

2 Related work

The World Wide Web (WWW) has considerably converted how data is consumed, generated, and processed. In the past decade, Web social media has undergone a boom as users are permitted to communicate themselves freely. With overwhelmingly generated short text documents on social media at an unprecedented high rate, the classic techniques to information retrieval (IR) and text mining were challenged to discover valuable information for user needs. Web Intelligence (WI) offers a new way to push the technology to manipulate the meaning of Web social media data and generate a distributed intelligence capable efficiently search engines [43, 44].

PRF using query expansion has been extensively employed in order to carry out research on social media search. It also presumed that it is valuable to make use of the most widely available terms in those documents with pseudo-relevance. A query expansion method has been proposed [28] regarding relevance feedback using the two-stage processing model. It

[1]The proposed model called query-based unsupervised short text mining (**QUSTM**)

focuses on searching by manually selecting tweets and integrating lexical evidence to builds relevance model. While it is noted that a user wishes to retrieve certain information for their needs, the proposed two-stage feedback model is a blend of different strategies, including language specific to the domain, the entity model, and the language collection model. In addition, [12] produce a separate entity model for every query-related entity that cannot represent the global semantics of the whole query.

The extant literature has suggested that the temporal prior has a powerful impact on the retrieval of information [8, 9, 35]. To investigate the link between time and relevance, a time-dependent language model was offered by [19]. It incorporates time into models that are both query-likelihood and relevance models. A temporal factor was suggested by [10] to smooth the language model and expand query by employing pseudo-relevance feedback and demonstrating its usefulness to recent searches. In microblogging media, [2] changed the use profile to deal with real-time filtering, thereby creating a balance between temporal interests for a specific topic. Miyanishi et al. [28] used two-stage pseudo-relevance feedback to examine the identical nature of recent profiles of the query and to prioritize the documents retrieved. In addition, the ranking function is another utility that incorporate temporal information. While related to the temporal queries (preference was given to recent retrievals), these models were adapted. More often, it appears to be a serious challenge to decide the unit used with time interval for various datasets and queries. [11] used kernel density projection and figured out the document's temporal prior instead of feedback documents. This approach was found to be more useful when re-ranking tweets and it used in this paper as baseline model to compare with the proposed model. It was also pointed out that there exists, independent of the content of documents, temporal signals. To rank them in order, [26] employed crowd signals and the temporal dynamic of query subtopics.

Machine learning has offered quite a few recent developments for overall social media search improvement [24, 26]. Learning to rank (L2R) is a field that takes advantage of this development. Three major categories can summarize the current state of work in L2R: pointwise, pairwise, and listwise. The major distinction exists in forming the problem and the veracious assumptions behind it as well as the spaces for input and/or output, and losing functions. The pointwise approach focuses on acquiring a relevance score for every query-document that a feature space represents [33]. The pairwise method focuses on learning whether to prioritize a query over a pair of documents [30]. Finally, the listwise method aims at finding the best-ranked list by directly enhancing the documents being input in to a query [23]. The prominent downside of L2R is its requirement for well-planned hand-based feature extraction; this can take a lot of time while still posing the risk of other problems.

The issue of social media related topic search is being addressed by a fast-developing research area. The very useful strength of this approach is helping us make effective searches within Social Media. Thus, tracing and gathering material linked to a specific topic requires the acquisition of new vocabulary and adjustments to be made to the primary representations of the topic at hand, as fresh but related sub-area are recognized. Topic discovery is another major issue in social media, as the text is quite short [17, 22, 31, 38]. The approach taken by works that address this problem is to aggregate short texts into pseudo-documents that are more lengthy [16, 27, 36].

LDA and pLSA, traditional topic models, are developed to openly focus on document-level and thereby capture the patterns of co-occurring words so that the structures of the topic can be discovered. Therefore, better and more reliable topical inferences will be possible as more word co-occurrences are carried out [13–15]. The short-text data sparsity issue has a considerable impact on traditional models, due to document length as results in infe-

rior inferences relating to a topic. External knowledge is from past research to further refine inferences based on short texts. Auxiliary long texts are used by [18] to infer dormant short text based topics and then cluster them. Such models need a bulky corpus of text of superior quality. Such texts may possibly not be available in some languages or domains. As short texts offers limited information, strategies that combine many short texts to generate pseudo-documents have been tried; these apply traditional approaches to modelling a topic in order to find out the hidden topics. The researchers in [36], combined many tweets by the same user to generate a pseudo-document prior to carrying out a conventional LDA strategy. Hashtags, name entities, and timestamps are other methods employed to aggregate short texts [16, 27, 41]. In some domains, good data is not available (e.g., in news headlines and in snippets from search). Thus, the literature informs us that it is imperative to design models specifically for short texts.

The significant difference between the proposed model and the above studies is the implicit relationships used to understand the social media short-text for a given query. The majority of the previous studies utilised lexical expansions from temporal evidence or external resources. In the proposed model, we obtain the lexical evidence for the user information needs based on the implicit relationships from local analysis regarding spareness issue. To the best of our knowledge, the proposed model is the first work to aggregate the short text tweets using query and infer the relationships in the new virtual document space to improve social media search without requiring external evidence from a knowledge base such as Wikipedia or Freebase.

3 Problem formulation

Given a query $Q = \{q_1, q_2, ..., q_m\}$ and a tweets collection C, an information retrieval system returns an initial ranked list of tweets $T = \{t_1, t_2, ..., t_k\}$ that contain k tweets. The proposed model utilises the top-k ranked tweets, which may include both relevant and non-relevant tweets for training the model. A tweet in the ranked list may be relevant to query Q; however, it maybe non-relevant to what users want. Let $\Omega = \{w_1, w_2, ..., w_n\}$ be a set of all terms in T where $w \in \Omega$ is a tweet token (e.g., a word). For each tweet $t_x \in T$, we assume there is a probability function $P_r : T \rightarrow [0, 1]$, which shows the probability of the tweet's relevance to what users want. For a given information retrieval system, which predicates the probability of relevance of tweets and sorts them in a ranked list, we have the following property:

$$P_r(t_1) \geq P_r(t_2) \geq \cdots \geq P_r(t_k)$$

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

4 The proposed model

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

4.1 Latent relationships

The main input for this phase is a set of retrieved tweets T for a given query Q which is used by a user to describe what she/he wants. Each tweet $t_x \in T$ is considered as an unlabelled tweet. In this paper, we use a language model "the query likelihood model with Drichlet smoothing" [39] to get the retrieved tweets T; this can be adapted for using with any retrieval model, such as BM25 [32], PTM [42], RFD [20]. Let Ω be a set of words (text features) for describing the relevant knowledge contained in retrieved tweets T. The objective here is to select Ω from T based on the query Q in order to describe the relevant to what the user wants.

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

$$\Gamma : Q \rightarrow 2^T \tag{1}$$

where mapping Γ can generate m sub-sets of tweets; we call each sub-set a *virtual document*.

Definition 1 (virtual document) : Let $Q = \{q_1, q_2, ..., q_m\}$ be a given query. A virtual document is a set of tweets that are related to an aspect of query Q. Formally, for each virtual document, there is a query term q_j, such that, the virtual document can be denoted as $\Gamma(q_j) = \{t_x | t_x \in T, q_j \in t_x\}$.

Table 1 shows an example of how to build virtual documents where the original query is $Q = \{q_1, q_2\}$ and a set of initial retrieved tweets is $T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$. In this example, we have two virtual documents (i.e., $\Gamma(q_1)$ and $\Gamma(q_2)$). A virtual document $\Gamma(q_1)$ includes

Table 1 An example of virtual documents

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

all tweets in the initial retrieved documents T that include query term q_1 and is defined as $\Gamma(q_1) = \{t_1, t_3, t_5\} = \{q_1, q_2, w_1, w_2, w_3, w_4, w_5, w_6, w_9, w_{10}, w_{11}, w_{12}\}$.

The rationale for making use of the virtual documents rather than original tweets when extracting informative features from the retrieved tweets is two-fold. First, in the above discussion, we use a mapping Γ to generate m virtual documents $\Gamma(q_j)$ for all $q_j \in Q$. For a given document, people (human beings) usually decide the relevance of the given document when reading through the whole document; in most cases, we say that document is relevant if we find a relevance sentence or a paragraph in the document. Thus, if any tweet $t_x \in \Gamma(q_j)$ is relevant, then we believe that $\Gamma(q_j)$ is relevant. Based on the above discussion, we can define:

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

Then, we can easily prove that

$$mean(P_r(\Gamma(q_j))) \geq mean(P_r(t_x))$$

This conclusion states that mapping Γ can reduce the extent of uncertainty in the retrieved tweets.

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

After obtaining the virtual documents for a set of retrieved tweets T, a new document space is introduced in which the relationships between Ω and the new document space should be investigated. Let $D = \{d_1, d_2, ..., d_m\}$ be the set of virtual documents $D = \{\Gamma(q_j)|q_j \in Q\}$. We can now obtain a one-one relation between Q and D, that is

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

We can also obtain a mapping between Ω and D, as follows:s

$$\xi : \Omega \rightarrow 2^D \tag{2}$$

where $\xi(w) = \{d_j|d_j \in D, w \in \Gamma(q_j)\}$. Figure 1 shows the relation between Q and Ω through the intermediate sets.

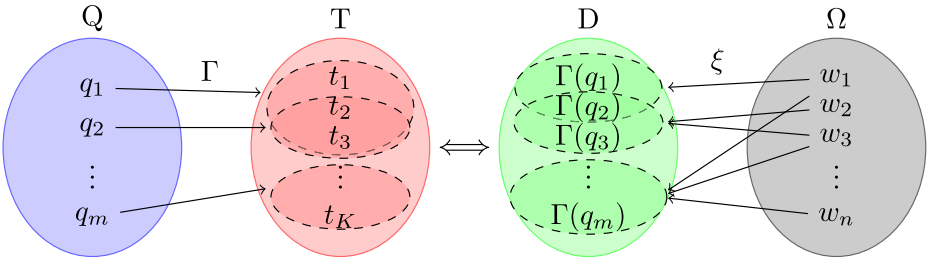

Figure 1 The relationship between Q, Ω and intermediate sets

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

$$R : \Omega \rightarrow 2^{Q} \tag{3}$$

where $R(w) = \{q_j | q_j \in Q, w \in \Gamma(q_j)\}$.

Figure 2 shows the relationships between Ω and Q in detail. The relationship includes P_q, a probability function for describing query terms' specificity, P_w, a probability function for describing the relevance of words to query Q and $g(w_i, q_j)$ which describes the strength of word w_i related to query term q_j.

4.2 Term estimation

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

$$Score(w_i) = P(w_i, D, Q) \cdot P_w(w_i) \tag{4}$$

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

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

$$\begin{aligned} P(w_i, D, Q) &= P(Q) \cdot P(w_i, D|Q) \\ &= P(Q) \cdot \sum_{q_j \in Q} \left[P(q_j) \cdot P(w_i, D|q_j) \right] \\ &\propto \sum_{j=1}^{m} P(w_i, D|q_j) \cdot P(q_j) \end{aligned} \tag{5}$$

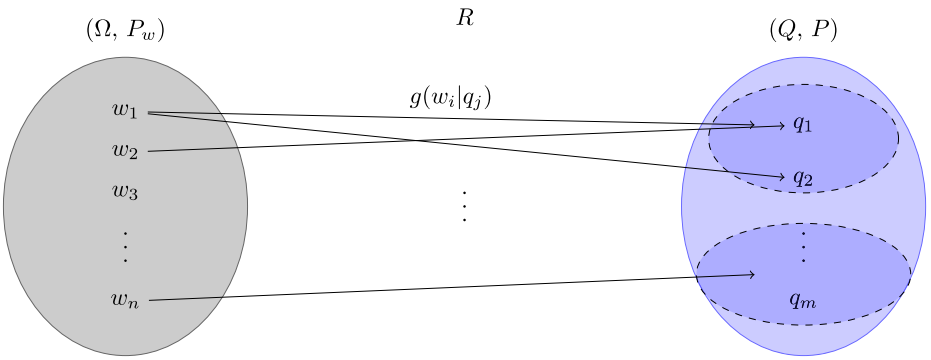

Figure 2 The relationship between Ω and Q via R

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

The following final estimation, for the score function $Score(w_i)$ of a words w_i, is given when we substitute (5) into (4):

$$Score(w_i) = P_w(w_i) \cdot \sum_{j=1}^{m} P(w, D|q_j) \cdot P(q_j) \tag{6}$$

In the implementation, we also give the following definitions for the concepts in Section 4.1. To instantiate the joint probability $P(w_i, D, Q)$ from (5), we estimated two main components: the word strength in a given virtual document $P(w_i, D|q_j)$ and the query terms' specificity $P(q_j)$. First, we estimate the strength of word w_i to query term q_j, as follows:

$$P(w_i, D|q_j) = g(w_i|q_j) = \frac{tf(w_i, q_j)}{|\Gamma(q_j)|} \tag{7}$$

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

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

$$P(q_j) = \frac{k - |\Gamma(q_j)|}{k} \tag{8}$$

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

We used $P_w(w_i)$ as a factor in the (6) to deal with the underlying uncertainty in the estimation of word relevance in virtual documents as the relevance documents is pseudo. We estimate the number of query terms $P_w(w_i)$ that map it in their virtual document between Q and Ω, as follows:

$$P_w(w_i) = |R(w_i)| = |\{q_j|q_j \in Q, w_i \in \Gamma(q_j)\}| \tag{9}$$

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

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

4.3 Algorithms

Algorithm 1 shows the proposed model framework where the input contains a set of retrieved tweets T, the original query Q and a word w_i in Ω. The algorithm starts with the initial steps for contracting the virtual documents from step 2 to step 8 where it aggregates all tweets in T that contain a given query term. Then, it uses the virtual documents to discover the implicit relationship between (Ω, P_w) and (Q, P) for a given word w_i. The algorithm used its a novel weighting schema. It starts from step 10-14 of the algorithm by verifying the given word w_i, followed by estimating the word w_i frequency in the current virtual document, as in (7), multiplied by the virtual document frequency, as in (8). Then, the algorithm continues for each virtual document $\Gamma(q_i)$ that contains a word w_i. Finally, we generalise a given word w_i based on how its frequent it is in the new space overall (in our case, in the virtual documents). Building the virtual documents can be done once, after which the weight for each word w_i in Ω can be estimated.

Algorithm 1 Score(w_i).

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

```
// construct virtual documents
foreach q_j ∈ Q do
    Γ(q_j) = ∅;
    foreach t_x ∈ T do
        if q_j ∈ t_x then
            Γ(q_j) = Γ(q_j) ∪ t_x;
        end
    end
end
// calculate P(w_i, D, Q) based on (6)
foreach q_j ∈ Q do
    estimate P(w_i, D|q_j) as in (7);
    estimate P(q_j) as in (8);
    w'_i = P(w_i, D|q_j) · P(q_j);
    P(w_i, D, Q) = P(w_i, D, Q) + w'_i;
end
calculate P_w(w_i) as in (9);
return P_w(w_i) · P(w_i, D, Q);
```

The time complexity of Algorithm 1 is determined by the "foreach" loops. The time complexity of the first "foreach" loop is $O(m \times k \times L)$ where L is the average size of a tweet. The time complexity of the second "foreach" loop depends on the process used when estimating $g(w_i|q_j)$, and the time complexity is $O(m \times S)$, where S is the average length of virtual document $\Gamma(q_j)$ and $S = O(k \times L)$. So, the time complexity is $O(m \times k \times L)$.

5 Experimental setup

5.1 Research questions.

RQ1: How does the proposed query-based unsupervised model perform compared to state-of-the-art algorithms in terms of improving social media search? **RQ2**: How does the proposed virtual documents schema improve the performance for other baseline models? **RQ3**: Is the proposed model performance sensitive to the number of selected tweets and number of terms utilised in the proposed model? **RQ4**: Is the proposed model performance stable across test sets?

5.2 Dataset.

In order to answer the research questions, we evaluate the proposed model for social media search application. Four standard Twitter test datasets from the TREC Microblog Tracks in 2011-2014 are selected [21, 29]. The TREC Microblog 2011-2012 dataset is called *Tweets2011* and had two query topics set in that name in article MB11 and MB12. The TREC Microblog 2013-2014 dataset is called *Tweets2013* and had two query topics set in that name in article MB13 and MB14. The size of the *Tweets2011* dataset is 16 million

Table 2 The statistics of the datasets

Topic set	MB11	MB12	MB13	MB14
No. of query topics	49	60	60	55
No. annotated feedbacks	35812	56512	60820	48386
No. of relevant feedbacks	2471	5381	8537	9710
No. of irrelevant feedbacks	33341	51131	52283	38676
Avg. length of query	3.43	2.86	3.30	3.78

tweets between January 23 and February 8, 2011 while that of *Tweets2013* is much larger, with 243 million tweets for the period February 1 to March 31, 2013. We utilised the official API[2] to crawl the datasets. Table 2 shows the statistics of the collections.

Pre-processing tweets was a critical stage to improve the retrieving model effectiveness [3]. The datasets pre-processing phase includes several strategies, as follows: (1) according to TREC microblog guidelines, we discarded non-English tweets using a language detector called ldig[3] and also discarded retweets; (2) we removed URLs from tweets as well as user mentions (i.e., "@user"); (3) since the hashtags (e.g., "#cnn") could include the query term, it treated them as normal tokens (e.g., "cnn"); (4) finally, we removed stop words, then stemmed tweet tokens using the Porter stemmer.

5.3 Experimental settings

The Dirichlet prior smoothing parameter μ sweeps over values from 50 to 1000 at an interval of 50. The number of tweets and the terms selected are set using two-fold cross-validation over each collection. We sweep the tweets' feedback between 10 to 100 with an interval of 10 and the selected number terms between 10 to 100 with a interval of 5. The parameters that are used in the baseline models, if required, are also set by utilising the same process.

5.4 Baselines

We compare the proposed model with a number of state-of-the-art baseline models. The baseline models are categorised into three groups: basic retrieval models, temporal models and topic models. The baseline models are as follows:

- LM.Dir a classical query likelihood language model with Dirichlet prior smoothing [39].
- BM25 a state-of-the-art probabilistic model that is utilised as retrieval model to find the similarity between a given query and collection [32].
- Recency a time-based language model that introduces a prior distribution for the document issued to which the recent documents for the given query are likely to be relevant [19].
- Kernel density estimation (KDE) one of the strongest time-based model which estimates the document time distribution using the kernel density [11].
- LDA a state-of-the-art topic model that finds the latent topics for a given collection [7].

[2] http://github.com/lintool/twitter-tools
[3] http://github.com/shuyo/ldig

Table 3 Comparison of the proposed method QUSTM and baselines models over MB2011 and MB2012 test sets where the highest value in each test set is marked in bold; superscripts 1,2,3 and 4 indicate statistically significant improvement at ($p < 0.05$) over LM.Dir, KDE and LDA; and the *ch%* line denotes the improvements over LM.Dir

	MB2011		MB2012	
Model	P@30	MAP	P@30	MAP
LM.Dir	0.3714	0.3561	0.3327	0.2248
BM25	0.3619	0.3504	0.3304	0.2222
Recency	0.3776	0.3581	0.3349	0.2255
KDE	0.3741	0.3398	0.3316	0.2249
RM3	0.3986	0.3712	0.3627	0.2534
LDA	0.3902	0.3347	0.3586	0.2353
PTM	0.3966	0.3506	0.3429	0.2466
QUSTM	**0.4102**^{1,2}	**0.4121**^{1,2,3}	**0.3870**^{1,2,3}	**0.2801**^{1,2,3}
ch%	+10.45%	+15.73%	+16.32%	+24.60%

– Pseudo document-based topic modelling (PTM) an innovative topic modelling approach that is designed for short-text analysis [45]. We exploited this as the proposed model in order to be fair; thus, the input for this model will be the number of top-ranked tweets. For the parameter settings, we followed the paper [34].
– RM3 [1] is a robust pseudo-relevance feedback method that estimates the relevance feedback using relevance models such as language model or BM25 and then, interpolates with the original query.

5.5 Evaluation metrics

The mean average precision (MAP) and the precision of cut-off n results (P@30) are used for evaluation. These metrics are the main evaluation metrics utilised in the Microblog tracks [21]. The statistical significance of differences in effectiveness is determined using two-sided paired t-tests at a 95% confidence level. The relevance judgments are assessed on a three-point scale: "not relevant", "relevant" and "highly relevant". In this paper, we followed [29] in terms of what we consider to be "highly relevant" as relevant.

5.6 Results and analysis

5.6.1 Overall performance

RQ1: We compare the proposed model performance with the baseline models. The results obtained by the proposed model and the baselines are presented in Tables 3 and 4. According to these tables, the temporal baseline approaches slightly outperform LM.Dir in most cases as well as the topic modelling approaches. In Tables 3 and 4, the proposed model outperforms all the baselines based on of P30 and MAP in all queries test sets over Tweets2011 and Tweets2013 collections. The statistical t-test shows that the P30 and MAP improvements over LM.Dir are significant in all queries test sets. These improvements over the strong baselines are also always significant. These results

Table 4 Comparison of the proposed method QUSTM and baselines models over MB2013 and MB2014 test sets where the highest value in each test set is marked in bold; superscripts 1,2,3 and 4 indicate statistically significant improvement at ($p < 0.05$) over LM.Dir, KDE and LDA; and the *ch%* line denotes the improvements over LM.Dir

	MB2013		MB2014	
Model	P@30	MAP	P@30	MAP
LM.Dir	0.4544	0.2825	0.6558	0.4573
BM25	0.4611	0.2803	0.6503	0.4523
Recency	0.4694	0.2875	0.6552	0.4606
KDE	0.4644	0.2791	0.6539	0.4641
RM3	0.4467	0.3035	0.6467	0.4951
LDA	0.4504	0.2755	0.6598	0.4771
PTM	0.4578	0.2864	0.6618	0.5031
QUSTM	**0.5422**^{1,2,3}	**0.3691**^{1,2,3}	**0.7000**^{1,2,3}	**0.5510**^{1,2,3}
ch%	+19.32%	+30.65%	+6.74%	+20.49%

show the effectiveness of the proposed method compared to state-of-the-art baselines approaches.

It can be clearly seen in the experiment that the proposed model outperformed and showed signicant improvement over the baseline models in all metrics across all microblog TREC dataset 2011-2014. Table 3 shows that for the MB11 test set, the proposed model improved over the P30 by a maximum improvement of 13.35% compared to BM25 and improved by a 2.91% minimum compared to RM3. The proposed model improved over the MAP by a maximum of 23.13% compared to LDA and a minimum of 11.02% compared to RM3. For the MB12 test set, the proposed model improved the P30 by a maximum of 17.13% compared to BM25 and a 6.70% minimum compared to RM3. The proposed model improved the MAP by a maximum of 26.06% against compared to BM25 and a 10.54% minimum against compared to RM3.

To confirm the superiority of the proposed model, we tested proposed model on the Tweets2013 dataset, which were much larger than the Tweets2011 dataset, and showed the variations in performance. For the MB13 test set, Table 4 shows that the proposed model improved the MAP by a maximum and minimum of 33.97% and 21.61% over LDA and RM3, respectively, while the corresponding increments of P30 were a maximum of 20.38% and minimum of 15.51% over LDA and Recency, respectively. For the MB14 test set, the proposed model improved the P30 by a maximum of 7.64% compared to BM25 and a 5.77% minimum compared to PTM. The proposed model improved the MAP by a maximum of 21.82% against compared to BM25 and a 9.52% minimum against compared to PTM.

Compared with the TREC Microblog tasks (see competition results in [21, 29]), the proposed model outperforms the majority of the best results. Specifically, for the MB11 test set, the proposed model improves MAP over the best submitted system of the task by 54.09%, while for the MB12 test set improves by 5.68%. The proposed model improves the MAP for the MB13 test set over the best submitted system by 4.52% while for the MB14 test set the proposed model decreases the MAP over the best system by 6.41%. In order to the best system for the MB14 test set, [25] employed the MB13 test set as training set via RankSVM and then utilised google search engine API to interpolate with local features,

Table 5 Results comparison

	MB2011		MB2012		MB2013		MB2014	
Model	P30	MAP	P30	MAP	P30	MAP	P30	MAP
LDA	0.3902	0.3347	0.3586	0.2353	0.4504	0.2755	0.6598	0.4771
VRLDA	0.4020	0.4037	0.3845	0.2787	0.4937	0.3247	0.6877	0.5401
QUSTM	**0.4102**	**0.4121**	**0.3870**	**0.2801**	**0.5422**	**0.3691**	**0.7000**	**0.5510**

whereas the proposed model does not use any external data. Also, for the MB14 test set, the proposed model improves the MAP over the TREC baseline by 43.92%.

5.6.2 Compared with LDA

RQ2: The proposed model has two major tasks: virtual document construction and term estimation. To verify the proposed query-based virtual document schema, we apply a

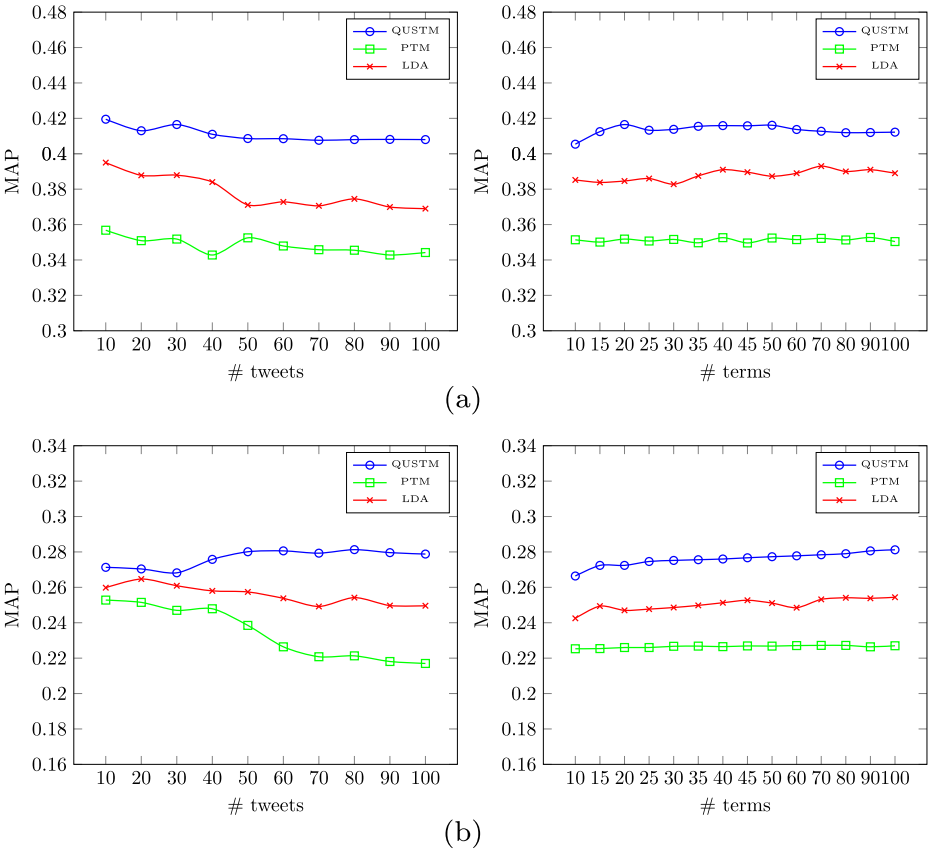

Figure 3 The proposed model performance in terms from a selected number of tweets T and terms Q' for all test sets. **a** MB11 and **b** MB12

virtual document to a state-of-the-art baseline model that is LDA. Table 5 show the proposed model QUSM compared with LDA and VRLDA (that stand virtual document + LDA) over both datasets in all test sets. First, in LDA, the main input is the original retrieved tweets where tweets are individuals. Then, to prove the effectiveness of the proposed virtual documents schema, we treat the input for LDA with our proposed virtual document schema. As Table 5 shows, VRLDA performance improved over P30 on average of 6.02% and significantly improved over MAP by a 17.53% compared to LDA. This significant improvement on LDA model when the main input is virtual documents indicates that there are high latent relationships between terms as we describe in Section 4.1. However, VRLDA still suffers to detect informative features that can describe the user information needs. The proposed model can reflect the discriminative for each candidate term in the virtual documents by estimate the accurate weight. As shown in Table 5, QUSTM significantly improved overall metrics and for all test set in both datasets.

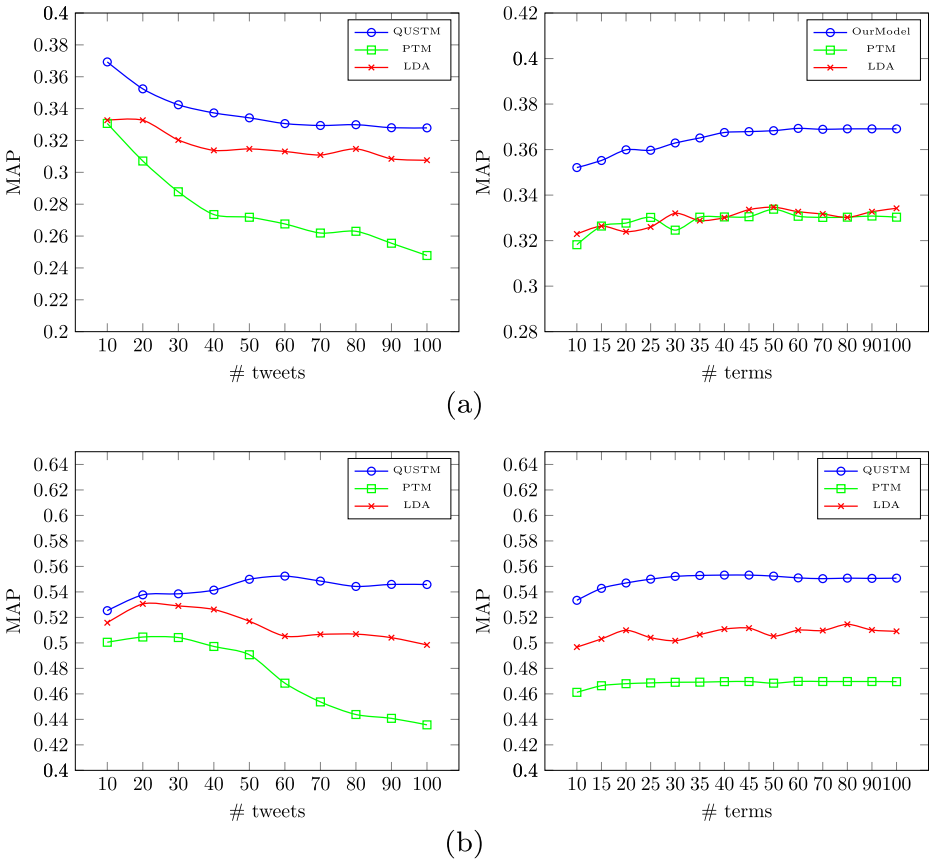

Figure 4 The proposed model performance in terms from a selected number of tweets T and terms Q' for all test sets. **a** MB13 and **b** MB14

5.6.3 The proposed model sensitivity.

RQ3: We show the proposed model sensitivity to the number of selected tweets k that represent the input of the proposed model and the number of selected terms from Ω in Figures 3 and 4. An important issue that could face the social media search system performance is the availability of relevant information in the selected tweets T and the terms. In order to maintain this issue, we investigate the proposed mode compared to the most robust baseline models that include LDA and PTM. Figures 3 and 4 show the MAP performance with a different number of k value and terms across all test sets over the Tweets2011 and Tweets2013 collections. The k value of tweets set T is tested from 10 to 100 with an interval value of 10, and the number of terms that are used as a new representation of Q' is set from 10 to 100 with an interval value of 5.

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

5.6.4 Per-query analysis.

RQ4: We conducted a comprehensive analysis of the improvements to the proposed model performance over the LM.Dir baseline on a per-query base. Figure 5 shows the per-query improvement histogram for the proposed model performance compared with the LM.Dir baseline model over 224 queries for all test sets over Tweets2011 and Tweets2013 collections. In practice, in MB11, the proposed model performance wins on 34 queries and loses on 13 queries out of 49 queries; in MB12, it wins on 50 queries and loses on 9 queries out of 60 queries. In MB13, the proposed model performance wins on 48 queries and loses on 12 queries out of 60 queries; in MB14, it wins on 41 queries and loses on 14 queries out of 55 queries. The average margin of the proposed model improvement is also greater than the losses, with 77%.

5.6.5 Efficiency analysis

As shown in Section 4.3, we analysis the time complexity of the proposed model. In this section, we show the practical implication of the proposed model. Figure 6 illustrate the computational latency of the proposed model against different number of top ranked tweets

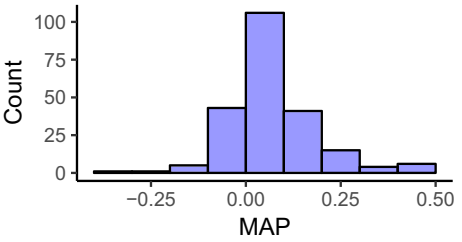

Figure 5 Difference in the MAP between the proposed model and LM.Dir using all test sets for both Tweets2011 and Tweets2013 datasets

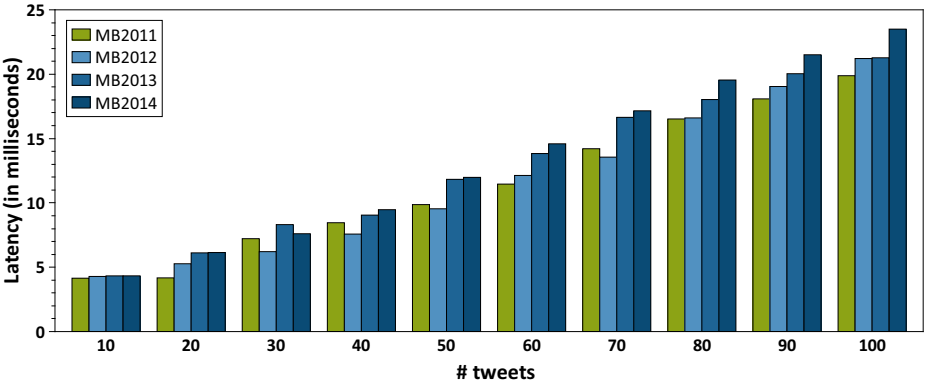

Figure 6 Average the real running time of the proposed method in milliseconds for all test queries

in the range 10 to 100 to all test quires sets (MB2011-2014). Note that[4], we compare running time on averaged over five runs for each query in the test set. The real running time estimate as in Algorithm 1 which start with building the virtual documents space to the end of the score function.

From the Figure 6, we observe that the running time of the proposed model between 4.148 and 23.5 millisecond on averaged for a query. The average time cost for a query on all test sets across different top ranked tweets set is 12.3686 millisecond.

6 Conclusion

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

In future work, we plan to integrate the temporal information with the evidence space in the proposed model. In addition, applying query performance predictor before applied the proposed model and interpolated with the number of user representation features. It interesting direction to set the first-pass retrieved documents dynamically for each user information need.

Acknowledgements This paper was partially supported by Grant DP140103157 from the Australian Research Council (ARC).

[4]The experiments are performed in a PC with an Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz and 16 GB memory running a Windows 7 operating system.

References

1. Abdul-Jaleel, N., Allan, J., Croft, W.B., Diaz, F., Larkey, L., Li, X., Smucker, M.D., Wade, C.: Umass at Trec 2004: Novelty and hard. In: TREC (2004)
2. Albakour, M., Macdonald, C., Ounis, I., et al.: On sparsity and drift for effective real-time filtering in microblogs. In: Proceedings of CIKM, pp 419–428 (2013)
3. Albishre, K., Albathan, M., Li, Y.: Effective 20 newsgroups dataset cleaning. In: Proceedings of WI-IAT, vol. 3, pp. 98-101 (2015)
4. Albishre, K., Li, Y., Xu, Y.: Effective pseudo-relevance for microblog retrieval. In: Proceedings of the Australasian Computer Science Week Multiconference, ACSW '17, pp. 51:1–51:6 (2017)
5. Albishre, K., Li, Y., Xu, Y.: Query-based automatic training set selection for microblog retrieval. In: Proceedings of PAKDD, pp. 325–336 (2018)
6. Atefeh, F., Khreich, W.: A survey of techniques for event detection in twitter. Comput. Intell. **31**(1), 132–164 (2015)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research **3**(Jan), 993–1022 (2003)
8. Chen, Q., Hu, Q., Huang, J., He, L.: Taker: Fine-grained time-aware microblog search with kernel density estimation. IEEE Transactions on Knowledge and Data Engineering (2018)
9. Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., Zheng, Z., Zha, H.: Time is of the essence: improving recency ranking using twitter data. In: Proceedings of WWW, pp. 331–340 (2010)
10. Efron, M., Golovchinsky, G.: Estimation methods for ranking recent information. In: Proceedings of SIGIR, pp. 495–504 (2011)
11. Efron, M., Lin, J., He, J., De Vries, A.: Temporal feedback for tweet search with non-parametric density estimation. In: Proceedings of SIGIR, pp. 33–42 (2014)
12. Fan, F., Qiangm, R., Lv, C., Yang, J.: Improving microblog retrieval with feedback entity model. In: Proceedings of CIKM, pp. 573–582 (2015)
13. Gao, Y., Xu, Y., Li, Y.: Pattern-based topics for document modelling in information filtering. IEEE Trans. Knowl. Data Eng. **27**(6), 1629–1642 (2015)
14. Gao, Y., Li, Y., Lau, R., Xu, Y., Bashar, M.: Finding semantically valid and relevant topics by association-based topic selection model. ACM Transactions on Intelligent Systems and Technology **9**(1), 3:1–3:22 (2017)
15. Gao, Y., Wenbo, W., Qian, L., Heyan, H., Li, Y.: Extending embedding representation by incorporating latent relations. IEEE Access **6**, 52682–52690 (2018)
16. Hong, L., Davison, B.D.: Empirical study of topic modeling in twitter. In: Proceedings of the first workshop on social media analytics, pp. 80–88 (2010)
17. Huang, J., Peng, M., Wang, H., Cao, J., Gao, W., Zhang, X.: A probabilistic method for emerging topic tracking in microblog stream. World Wide Web **20**(2), 325–350 (2017)
18. Jin, O., Liu, N.N., Zhao, K., Yu, Y., Yang, Q.: Transferring topical knowledge from auxiliary long texts for short text clustering. In: Proceedings of CIKM, pp. 775–784 (2011)
19. Li, X., Croft, W.B.: Time-based language models. In: Proceedings of CIKM, pp. 469–475 (2003)
20. Li, Y., Algarni, A., Albathan, M., Shen, Y., Bijaksana, M.A.: Relevance feature discovery for text mining. IEEE Transactions on Knowledge and Data Engineering **27**(6), 1656–1669 (2015)
21. Lin, J., Efron, M.: Overview of the trec-2013 microblog track. In: TREC, pp. 1–5 (2013)
22. Lin, T., Tian, W., Mei, Q., Cheng, H.: The dual-sparse topic model: mining focused topics and focused terms in short text. In: Proceedings of WWW, pp. 539–550 (2014)
23. Liu, S., Cheng, X., Li, F.: Ranking tweets by labeled and collaboratively selected pairs with transitive closure. In: Proceedings of AAAI, pp. 1235–1241 (2014)
24. Luo, Z., Osborne, M., Wang, T.: An effective approach to tweets opinion retrieval. World Wide Web **18**(3), 545–566 (2015)
25. Lv, C., Fan, F., Qiang, R., Fei, Y., Yang, J.: Pkuicst at trec 2014 Microblog Track: Feature Extraction for Effective Microblog Search and Adaptive Clustering Algorithms for Ttg. In: TREC (2014)
26. Martins, F., Magalhães, J., Callan, J.: Barbara made the news: Mining the behavior of crowds for time-aware learning to rank. In: Proceedings of WSDM, pp. 667–676 (2016)
27. Mehrotra, R., Sanner, S., Buntine, W., Xie, L.: Improving lda topic models for microblogs via tweet pooling and automatic labeling. In: Proceedings of SIGIR, pp. 889–892 (2013)
28. Miyanishi, T., Seki, K., Uehara, K.: Improving pseudo-relevance feedback via tweet selection. In: Proceedings of CIKM, pp. 439–448 (2013)
29. Ounis, I., Macdonald, C., Lin, J., Soboroff, I.: Overview of the trec-2011 microblog track. In: TREC, vol. 32 (2011)

30. Qiang, R., Liang, F., Yang, J.: Exploiting ranking factorization machines for microblog retrieval. In: Proceedings of CIKM, pp. 1783–1788 (2013)
31. Ramage, D., Dumais, S.T., Liebling, D.J.: Characterizing microblogs with topic models. In: Proceedings of AAAI, vol. 10, p. 16 (2010)
32. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M.M., Gatford, M., et al.: Okapi at trec-3. Nist Special Publication **109**, 109 (1995)
33. Severyn, A., Moschitti, A., Tsagkias, M., Berendsen, R., De Rijke, M.: A syntax-aware re-ranker for microblog retrieval. In: Proceedings of SIGIR, pp. 1067–1070 (2014)
34. Shi, T., Kang, K., Choo, J., Reddy, C.K.: Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In: Proceedings of WWW, pp. 1105–1114 (2018)
35. Wang, Y., Huang, H., Feng, C.: Query expansion based on a feedback concept model for microblog retrieval. In: Proceedings of WWW, pp. 559–568 (2017)
36. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of WSDM, pp. 261–270 (2010)
37. Wu, S., Huang, C.: Search result diversification via data fusion. In: Proceedings of SIGIR, pp. 827–830 (2014)
38. Yan, X., Guo, J., Lan, Y., Cheng, X.: A biterm topic model for short texts. In: Proceedings of WWW, pp. 1445–1456 (2013)
39. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of SIGIR, pp. 334–342 (2001)
40. Zhang, Z., Wang, Q., Si, L., Gao, J.: Learning for efficient supervised query expansion via two-stage feature selection. In: Proceedings of SIGIR, pp. 265–274 (2016)
41. Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H., Li, X.: Comparing twitter and traditional media using topic models. In: Proceedings of ECIR, pp. 338–349 (2011)
42. Zhong, N., Li, Y., Wu, S.T.: Effective pattern discovery for text mining. IEEE Trans. Knowl. Data Eng. **24**(1), 30–44 (2012)
43. Zhong, N., Liu, J., Yao, Y.: Web intelligence. Springer Science & Business Media (2013)
44. Zhong, N., Liu, J., Shi, Y., Yao, Y.: An interview with professor raj reddy on Web intelligence (wi) and computational social science (css). Web Intelligence **16**(3), 143–146 (2018)
45. Zuo, Y., Wu, J., Zhang, H., Lin, H., Wang, F., Xu, K., Xiong, H.: Topic modeling of short texts: A pseudo-document view. In: Proceedings of KDD, pp. 2105–2114 (2016)

Affiliations

Khaled Albishre[1,2] · Yuefeng Li[1] · Yue Xu[1] · Wei Huang[3]

Yuefeng Li
y2.li@qut.edu.au

Yue Xu
yue.xu@qut.edu.au

[1] School of EECS, Queensland University of Technology (QUT), Brisbane, Australia

[2] Umm Al-Qura University, Makkah, Saudi Arabia

[3] School of Economy and Management, Hubei University of Technology, Wuhan, 430064, Hubei China