Front. Comput. Sci., 2018, 12(4): 714–724
https://doi.org/10.1007/s11704-016-6163-5

RESEARCH ARTICLE

Learning from context: a mutual reinforcement model for Chinese microblog opinion retrieval

Jingjing WEI[1,2,3], Xiangwen LIAO (✉)[1,3], Houdong ZHENG[1,3], Guolong CHEN[1,3], Xueqi CHENG[4]

1 College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China
2 College of Electronics and Information Science, Fujian Jiangxia University, Fuzhou 350108, China
3 Fujian Provincial Key Laboratory of Network Computing and Intelligence Processing, Fuzhou University, Fuzhou 350108, China
4 Institute of Computing Technology, Chinese Academic of Sciences, Beijing 100190, China

Abstract This study addresses the problem of Chinese microblog opinion retrieval, which aims to retrieve opinionated Chinese microblog posts relevant to a target specified by a user query. Existing studies have shown that lexicon-based approaches employed online public sentiment resources to rank sentiment words relying on the document features. However, this approach could not be effectively applied to microblogs that have typical user-generated content with valuable contextual information: "user–user" interpersonal interactions and "user–post/comment" intrapersonal interactions. This contextual information is very helpful in estimating the strength of sentiment words more accurately. In this study, we integrate the social contextual relationships among users, posts/comments, and sentiment words into a mutual reinforcement model and propose a unified three-layer heterogeneous graph, on which a random walk sentiment word weighting algorithm is presented to measure the strength of opinion of the sentiment words. Furthermore, the weights of sentiment words are incorporated into a lexicon-based model for Chinese microblog opinion retrieval. Comparative experiments are conducted on a Chinese microblog corpus, and the results show that our proposed mutual reinforcement model achieves significant improvement over previous methods.

Keywords opinion retrieval, sentiment words, lexicon weighting, mutual reinforcement model

1 Introduction

With the rapid growth of Web 2.0 technologies, microblogging has become an important form of communication. It is very convenient for people to share personal attitudes and feelings with others by publishing brief posts and comments on microblog platforms such as Twitter and Sina Weibo. These opinions are very helpful for other users to make decisions and obtain valuable feedback, and could be widely used in business intelligent applications such as product surveys, recommendation systems, and advertisement analysis [1]. Therefore, opinion mining research, including opinion retrieval, sentiment classification, and opinion summarization, has attracted considerable interest from both academic researchers and commercial companies [2–5]. This study focuses on the problem of Chinese microblog opinion retrieval, which aims at finding people's opinions or subject attitudes towards some objects from user-generated data on a Chinese microblog platform. The fundamental challenge of the problem is how to identify those query-related opinions, and then design an optimal model to combine the topical relevance and opinion to produce a single ranking.

Currently, several related works [5–10] considered opin-

Received March 20, 2016; accepted August 22, 2016

E-mail: liaoxw@fzu.edu.cn

ion retrieval as a two-stage method. In the first stage, relevant documents are retrieved by their relevance scores (e.g., term frequency–inverse document frequency (TF–IDF) value or language model). In the second stage, an opinion score (the ranking score of to what extent it is subjective or objective) is calculated for each retrieved document by using machine learning methods, lexicon-based statistical approaches, or some other methods. Finally, an overall score combining the two scores is computed to re-rank the retrieved documents. Most existing approaches have employed a linear combination of relevance score and opinion score.

Nevertheless, the two-stage methods suffer from a lack of adequate theoretical support and, hence, unified opinion retrieval models were proposed [11–17]. Typical unified models, such as generative models [12], topic models [14], and learning to rank [15], directly mine the relevant opinions for documents ranking and, thus, have the advantage of expressing the users' information requirements more straightforwardly and effectively. However, most existing approaches for calculating opinion scores have been based on sentiment lexicon, which was assumed to be uniformly distributed or only weighted according to the document information on an external dataset. In fact, it is not reasonable that different sentiment words in one document have the same opinion strength or that the same sentiment word from different documents has the same opinion strength. For example, "unpredictable" is negative in an electronics document while it is positive in a movie document. Microblog posts are potentially networked through user connections, which may contain useful semantic clues that are not available in the purely short posts. Moreover, microblogs are user-generated content and contain rich user contextual interaction information that would benefit from text-mining tasks such as social contextual recommendation [18], sentiment lexicon construction [19], and sentiment classification [20], and this contextual information is very helpful for weighting sentiment words.

In this paper, we propose a novel mutual reinforcement (MR) model for ranking sentiment words that incorporates user-level contextual relationships: "user–user" interpersonal interactions and "user–post/comment" intrapersonal interactions. First, the social contextual relationships among users, posts/comments, and sentiment words are used to construct a unified three-layer heterogeneous graph, on which a random walk sentiment word weighting algorithm is presented to measure opinion strength of the sentiment words more precisely. Then, the weights of sentiment words resulting from our MR model are incorporated into two typical unified retrieval models (i.e., generative model and learning to rank) for Chinese microblog opinion retrieval. Finally, a real dataset is constructed from the Chinese microblog platform Sina Weibo and experimental results show that when evaluated on the real dataset, the proposed model can achieve more effective opinion retrieval performance compared with two other state-of-the-art opinion retrieval methods. The main contributions of the paper are three-fold.

- A MR framework is presented to incorporate microblog user-level contextual relationships (i.e., "user–user" interpersonal interactions and "user–post/comment" intrapersonal interactions) for sentiment word weighting.
- A solid mathematical description for the three-level (i.e., user– post/comment–sentiment words) MR ranking is provided and the convergence of the algorithm is guaranteed.
- The effectiveness of the proposed three-level MR sentiment word ranking algorithm is examined in the context of Chinese microblog opinion retrieval.

The rest of this paper is organized as follows. In Section 2, we review related work on opinion retrieval. Section 3 presents our MR model for opinion retrieval. We evaluate our model with comparative experiments and the results are presented in Section 4. Finally, in Section 5, the paper is concluded and future work is suggested.

2 Related work

2.1 Two-stage model

Opinion retrieval has attracted much attention recently. Many research institutions have presented work on opinion retrieval, such as the Text Retrieval Conference (TREC), NII Testbeds and Community for Information access Research (NTCIR) Project, and the Chinese Opinion Analysis Evaluation (COAE) [2–4]. Previous work has adopted a two-stage approach. In the first stage, this approach mostly uses a classic retrieval model such as a language model or BM25 model. Most research focuses on the second stage, that is, opinion identification. Zhang et al. proposed a three-component opinion retrieval algorithm [6]. They retrieved the relevant documents in the first component. Then, a support vector machine (SVM) classifier was built to identify opinionative documents in the second stage. Finally, multiple strategies were used to rank the retrieved relevant opinionative documents in the third component. Santos et al. proposed a novel approach that integrated the proximity of query terms to subjective sentences identified from the retrieved documents for blog opinion retrieval [7]. The divergence from randomness (DFR) proximity model was extended to integrate the proximity of query terms to the subjective sentences that were identified by two opinion-detection techniques. Jiang et al. used both target-dependent and context-aware approaches for target-dependent sentiment classification of tweets [8]. Wang et al. addressed the task of hashtag-level sentiment classification and proposed a graph model that effectively incorporated the sentiment information from tweets, the literal meaning of hashtags, and hashtags co-occurrence relationships to tackle the task [9]. Hu et al. studied the problem of interpreting emotional signals for unsupervised sentiment analysis [10]. To verify the existence of emotional signals, the exploratory study was conducted on two Twitter datasets via statistical hypothesis testing first, and then the signals were incorporated into an unsupervised learning framework for sentiment analysis.

2.2 Unified opinion retrieval model

As the two-stage model suffers from the lack of adequate theoretical support, the unified opinion retrieval model has become a topic of considerable research interest. Eguchi and Lavrenko proposed a probabilistic generative language model for opinion retrieval at the sentence level [11]. The sentiment relevance models and topic relevance models were combined into a unified model considering the topic dependence of the sentiment. Zhang and Ye focused on retrieving relevant opinions over general topics and proposed a formal probabilistic generation model that unifies topic relevance and lexicon-based sentiment [12]. Huang and Croft paid attention to the problem of representing the information need for opinion retrieval [13]. They proposed an opinion relevance model to represent the information need. Then a unified opinion retrieval model was developed by computing the Kullback–Leibler (KL) divergence between the two probability distributions of the opinion relevance model and document model. Mei et al. proposed a new probabilistic topic–sentiment mixture (TSM) model that could learn general sentiment models and extract topic models for topic–sentiment analysis on Weblogs [15]. The topic life cycles and the associated sentiment dynamics could also be extracted by this model.

Some efforts have also been made to explore other methods. Luo et al. proposed a ranking model for opinion retrieval in Twitter [14]. They incorporated social and opinionated information into a learning-to-rank (LTR) model for tweets opinion retrieval.

2.3 Social contextual information for text mining

Recently, contextual information in social media such as microblogs has been effectively exploited to enhance the performance of text-mining algorithms. Li et al. [5] constructed a new information representation, namely topic–sentiment word pairs, to capture intra-sentence and inter-sentence contextual information. Finally, a unified graph-based model was proposed to rank documents for opinion retrieval by combining the two types of contextual information. Jiang et al. [18] proposed a social contextual recommendation framework based on a probabilistic matrix factorization method to incorporate individual preference and interpersonal influence to improve the accuracy of social recommendation. Feng et al. [19] integrated the emoticons and candidate sentiment words in the microblogs to construct a two-layer graph, on which a random walk is run to extract the top-ranked words as a sentiment lexicon. Li et al. [21] proposed a novel learning framework to exploit the reviewer and product information for review rating prediction. A tensor-based method was presented to represent the relationship among reviewers, products, and text features. Hu et al. [20] proposed a novel sociological approach (SANT) that can utilize sentiment relations between messages to facilitate sentiment classification and effectively handle noisy Twitter data.

In addition to the document information, social media channels such as microblogs have social context information that can be used to estimate the strength of sentiment words more precisely. Different from previous works [18–22], our proposed model exploits more user-level social contextual information, and then integrates the contextual information with document information for enhancing the performance of Chinese microblog opinion retrieval.

3 Proposed methods

In this section, we first introduce the MR model that explicitly integrates "user–user" interpersonal relationships and "user–post/comment" intrapersonal relationships into a unified ranking framework. Second, the convergence of our proposed MR model is investigated. Finally, we describe how opinion retrieval can be improved by a MR model.

3.1 MR model for sentiment lexicon weighting

In this section, a three-layer MR graph is constructed to take advantage of the homogeneous relationships among users, posts/comments, and sentiment words. Then, based on the heterogeneous graph, a unified random walk sentiment word weighting algorithm is present to capture more precisely the strength of the sentiment words. Figure 1 shows an overview of the heterogeneous graph.

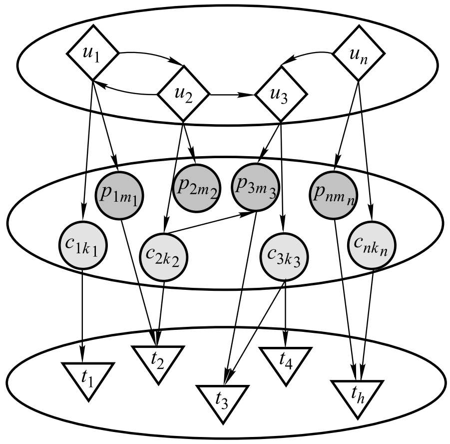


Fig. 1 The MR graph

Without loss of generality, let $U = \{u_1, u_2, \ldots, u_n\}$ be a user set, $P = \{p_{11}, \ldots, p_{1m_1}, \ldots, p_{n1}, \ldots, p_{nm_n}\}$ be a post set, $C = \{c_{11}, \ldots, c_{1k_1}, \ldots, c_{n1}, \ldots, c_{nk_n}\}$ be a comment set, and $T = \{t_1, t_2, \ldots, t_h\}$ be a sentiment word set. In the user level, if user u_i follows user u_j, a direct edge from u_i to u_j is created. When user u_i published post p_{ik} or c_{im}, we connect u_i with p_{ik} or c_{im}. In the post/comment level, if a user u_i writes a comment c_{im} for a post p_{jk}, then c_{im} is linked to p_{jk}. An edge from a post or comment to a sentiment word t_j denotes that this sentiment word is included in that post or comment. As a result, the MR graph is formed with three layers for users, posts/comments, and sentiment words. The strength of sentiment words can be ranked according to the MR effects between users, posts/comments, and sentiment words. Table 1 provides a list of variables.

The MR model over the heterogeneous graph is proposed based on the following MR principles.

1) The authority of a user is high if: (i) the user published posts or comments with high quality; (ii) they use important words; (iii) they are followed by other high-authority users. The ranking score of a user is defined as

$$\begin{aligned} Aur^{(r+1)}(u_i) = \alpha_1 \sum_{u \in Fol_{u_i}} \frac{1}{|Fol_{u_i}|} Aur^{(r)}(u) \\ +\beta_1 [\sum_{p \in P_{u_i}} Qual^{(r)}(p) + \sum_{c \in C_{u_i}} Qual^{(r)}(c)] \\ +\gamma_1 \sum_{t \in T_{u_i}} \mathrm{Im}p^{(r)}(t). \end{aligned} \tag{1}$$

2) The quality of a post is high if: (i) it is published by a high-authority user; (ii) it is associated with high-quality comments; (iii) it contains important sentiment words. The ranking score of a post is written as

$$\begin{aligned} Qual^{(r+1)}(p_{ij}) = \alpha_2 Aur^{(r)}(u_i) + \beta_2 \sum_{c \in C_{p_{ij}}} Qual^{(r)}(c) \\ +\gamma_2 \sum_{t \in T_{p_{ij}}} \mathrm{Im}p^{(r)}(t). \end{aligned} \tag{2}$$

3) A comment is of high quality if it is published by a high-authority user, and contains important sentiment words. The ranking score of a comment is described as

$$Qual^{(r+1)}(c_{ij}) = \alpha_3 Aur^{(r)}(u_i) + \gamma_3 \sum_{t \in T_{c_{ij}}} \mathrm{Im}p^{(r)}(t). \tag{3}$$

4) A sentiment word is important if it is used by high-authority users and contained in high-quality posts and comments. The ranking score of a sentiment word is defined as

$$\begin{aligned} \mathrm{Im}p^{(r+1)}(t_i) = \beta_4 [\sum_{p \in P_{t_i}} \frac{1}{|P_{t_i}|} Qual^{(r)}(p) \\ + \sum_{c \in C_{t_i}} \frac{1}{|C_{t_i}|} Qual^{(r)}(c)] \\ +\alpha_4 \sum_{u \in U_{t_i}} \frac{1}{|U_{t_i}|} Aur^{(r)}(u). \end{aligned} \tag{4}$$

Table 1 Summary of symbols used in the paper

Symbol	Description
r	The rth iteration
$Aur^{(r)}(u)$	The ranking score of user u
$Qual^{(r)}(p)$	The ranking score of post p
$Qual^{(r)}(c)$	The ranking score of post comment c
$\mathrm{Im}p^{(r)}(t)$	The ranking score of sentiment word t
Fol_{u_i}	The set of users following u_i
P_{u_i}	The set of posts published by u_i
C_{u_i}	The set of comments published by u_i
T_{u_i}	The set of sentiment words used by u_i
$C_{p_{ij}}$	The comments of post p_{ij}
$T_{p_{ij}}$	The words p_{ij} contains
$T_{c_{ij}}$	The words c_{ij} contains
U_{t_i}	Users using t_i
P_{t_i}	Posts containing t_i
C_{t_i}	Comments containing t_i

3.2 Convergence of our MR methods

As one of the fundamental problems of the iteration algorithm, the convergence of the presented algorithm is investigated in this section. Inspired by [22], we formulate our MR method in the following format:

$$\begin{cases} R_U^{(r+1)} = \alpha_1 U_U R_U^{(r)} + \beta_1 U_P R_P^{(r)} + \beta_1 U_C R_C^{(r)} + \gamma_1 U_T R_T^{(r)}, \\ R_P^{(r+1)} = \alpha_2 P_U R_U^{(r)} + \beta_2 P_P R_P^{(r)} + \beta_2 P_C R_C^{(r)} + \gamma_2 P_T R_T^{(r)}, \\ R_C^{(r+1)} = \alpha_3 C_U R_U^{(r)} + \beta_3 C_P R_P^{(r)} + \beta_3 C_C R_C^{(r)} + \gamma_3 C_T R_T^{(r)}, \\ R_T^{(r+1)} = \alpha_4 T_U R_U^{(r)} + \beta_4 T_P R_P^{(r)} + \beta_4 T_C R_C^{(r)} + \gamma_4 T_T R_T^{(r)}, \end{cases} \tag{5}$$

where U_U denotes the U–U affinity matrix, U_P denotes the U–P affinity matrix, U_C denotes the U–C affinity matrix, U_T denotes the U–T affinity matrix, and so on. Here

$$W = \begin{bmatrix} \alpha_1 & \beta_1 & \beta_1 & \gamma_1 \\ \alpha_2 & \beta_2 & \beta_2 & \gamma_2 \\ \alpha_3 & \beta_3 & \beta_3 & \gamma_3 \\ \alpha_4 & \beta_4 & \beta_4 & \gamma_4 \end{bmatrix},$$

is the weight matrix for balancing the relative weight of the user, post, comment, and sentiment word.

The coefficients in Eq. (5) correspond to a block matrix,

$$M = \begin{bmatrix} \alpha_1 U_U & \beta_1 U_P & \beta_1 U_C & \gamma_1 U_T \\ \alpha_2 P_U & \beta_2 P_P & \beta_2 P_C & \gamma_2 P_T \\ \alpha_3 C_U & \beta_3 C_P & \beta_3 C_C & \gamma_3 C_T \\ \alpha_4 T_U & \beta_4 T_P & \beta_4 T_C & \gamma_4 T_T \end{bmatrix}. \tag{6}$$

From the above relationships, we know that P_P, C_P, C_C, and T_T are all zero matrices.

Let

$$R = \begin{bmatrix} R_U \\ R_P \\ R_C \\ R_T \end{bmatrix}.$$

To make R converge to a stationary vector, we must force M to be stochastic and irreducible using the following steps: (i) split M into four matrices by column, and let X denote any of the four diagonal block matrices, such as

$$\begin{bmatrix} U_U \\ P_U \\ C_U \\ T_U \end{bmatrix}, \begin{bmatrix} U_P \\ P_P \\ C_P \\ T_P \end{bmatrix}, \begin{bmatrix} U_C \\ P_C \\ C_C \\ T_C \end{bmatrix}, \text{and} \begin{bmatrix} U_T \\ P_T \\ C_T \\ T_T \end{bmatrix};$$

(ii) find the columns that contain all zero elements in X, and replace these columns by $\vec{e}/k$ ($\vec{e}$ is the column vector of all ones and k is the line order of the matrix $\overline{X}$); (iii) replace X by $\overline{X}$ in M, and let $\overline{M}$ replace M.

Since the edges in the graph represent the interactions among the user, post/comment, and sentiment word, the graph ensures that $W \geqslant 0$. It is easy to make W column stochastic, and we can prove $\overline{M}$ is a stochastic matrix as follows.

Theorem 1 We have that $\overline{M}$ is column stochastic.

Proof Let $A^{(k)}$, $B^{(k)}$, $C^{(k)}$, and $D^{(k)}$ denote the four blocks in the column of $\overline{M}$ ($k = 1, 2, 3, 4$). Here α_1–α_4, β_1–β_4, and γ_1–γ_4 are the corresponding weight coefficients. Then the weight matrix W meets the following conditions:

$$\begin{cases} \sum_i \overline{X}_{ij}^{(1)} = \alpha_1 \sum_i A_{ij}{}^{(1)} + \alpha_2 \sum_i B_{ij}{}^{(1)} + \alpha_3 \sum_i C_{ij}{}^{(1)} \\ \qquad +\alpha_4 \sum_i D_{ij}{}^{(1)} = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1, \\ \sum_i \overline{X}_{ij}^{(2)} = \beta_1 \sum_i A_{ij}{}^{(2)} + \beta_2 \sum_i B_{ij}{}^{(2)} + \beta_3 \sum_i C_{ij}{}^{(2)} \\ \qquad +\beta_4 \sum_i D_{ij}{}^{(2)} = \beta_1 + \beta_2 + \beta_3 + \beta_4 = 1, \\ \sum_i \overline{X}_{ij}^{(3)} = \beta_1 \sum_i A_{ij}{}^{(3)} + \beta_2 \sum_i B_{ij}{}^{(3)} + \beta_3 \sum_i C_{ij}{}^{(3)} \\ \qquad +\beta_4 \sum_i D_{ij}{}^{(3)} = \beta_1 + \beta_2 + \beta_3 + \beta_4 = 1, \\ \sum_i \overline{X}_{ij}^{(4)} = \gamma_1 \sum_i A_{ij}{}^{(4)} + \gamma_2 \sum_i B_{ij}{}^{(4)} + \gamma_3 \sum_i C_{ij}{}^{(4)} \\ \qquad +\gamma_4 \sum_i D_{ij}{}^{(4)} = \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 = 1. \end{cases} \tag{7}$$

Next, $\overline{M}$ is made irreducible. According to the PageRank model, a surfer accesses a page with a tiny probability from one page to another. Thus, one more adjustment, to make P irreducible, is implemented:

$$\overline{X}' = b\overline{X} + (1-b)A/n, \tag{8}$$

where A is a matrix of all ones, n is the line order of the matrix $\overline{X}$, and b is set to 0.85 in reference to PageRank. We finally replace $\overline{X}$ by $\overline{X}'$ in $\overline{M}$, and let $\overline{M}'$ replace $\overline{X}'$. □

Theorem 2 We have that $\overline{M}'$ is irreducible.

Proof In the graph $\overline{M}'$, let P_1, P_2, P_3, P_4 be the graph corresponding to the diagonal block matrix U_U, P_P, C_C, T_T. Every node is now directly connected to every other node in P_1. Meanwhile, we note that for any one node n_i in P_1, there exists at least one node n_j in P_2 linking n_i. In the same way, there exists at least one node n_k in P_3 linking one node n_j in P_2, and at least one node n_l in P_4 linking one node n_k in P_3, so the graph corresponding to $\overline{M}'$ is strongly connected. Thus, $\overline{M}'$ must be irreducible. □

As a result, $\overline{M}'$ is stochastic and irreducible. Since each element is nonnegative in $\overline{M}'$, $\overline{M}'$ is primitive and the spectral radius $\rho(\overline{M}')$ is 1. The eigenvector of $\overline{M}'$ can be computed when the eigenvalue is equal to 1. Thus, the power method will converge to the stationary vector R:

$$\overline{M}'R = R. \tag{9}$$

Finally, an iterative algorithm can be developed to solve

Eq. (5). For convenience in the algorithm, $\overline{M}'$ is rewritten as

$$G = \begin{bmatrix} \alpha_1 G_{11} & \beta_1 G_{12} & \beta_1 G_{13} & \gamma_1 G_{14} \\ \alpha_2 G_{21} & \beta_2 G_{22} & \beta_2 G_{23} & \gamma_2 G_{24} \\ \alpha_3 G_{31} & \beta_3 G_{32} & \beta_3 G_{33} & \gamma_3 G_{34} \\ \alpha_4 G_{41} & \beta_4 G_{42} & \beta_4 G_{43} & \gamma_4 G_{44} \end{bmatrix}. \tag{10}$$

The details of algorithm are described in the following.

Algorithm Obtain the unique eigenvector R

Input: $R^{(0)}$ is a random vector and $|s| = 1$.

Output: $R^{(r)}$

1: $r \leftarrow 0, \eta$;

2: **while** $(\eta < 10^{-3})$;

3:

$$\begin{cases} R_U^{(r+1)} = \alpha_1 G_{11} R_U^{(r)} + \beta_1 G_{12} R_P^{(r)} + \beta_1 G_{13} R_C^{(r)} \\ \quad + \gamma_1 G_{14} R_T^{(r)}, \\ R_P^{(r+1)} = \alpha_2 G_{21} R_U^{(r)} + \beta_2 G_{22} R_P^{(r)} + \beta_2 G_{23} R_C^{(r)} \\ \quad + \gamma_2 G_{24} R_T^{(r)}, \\ R_C^{(r+1)} = \alpha_3 G_{31} R_U^{(r)} + \beta_3 G_{32} R_P^{(r)} + \beta_3 G_{33} R_C^{(r)} \\ \quad + \gamma_3 G_{34} R_T^{(r)}, \\ R_T^{(r+1)} = \alpha_4 G_{41} R_U^{(r)} + \beta_4 G_{42} R_P^{(r)} + \beta_4 G_{43} R_C^{(r)} \\ \quad + \gamma_4 G_{44} R_T^{(r)}; \end{cases};$$

4:

$$\eta \leftarrow \max\left[\begin{Bmatrix} \left\|R_U^{(r+1)} - R_U^{(r)}\right\|_2^2 \\ \left\|R_P^{(r+1)} - R_P^{(r)}\right\|_2^2 \\ \left\|R_C^{(r+1)} - R_C^{(r)}\right\|_2^2 \\ \left\|R_T^{(r+1)} - R_T^{(r)}\right\|_2^2 \end{Bmatrix}\right];$$

5:

$$r \leftarrow r + 1;$$

6: **End of while;**

7: **Return** $R^{(r)} = [R_U^{(r)} R_P^{(r)} R_C^{(r)} R_T^{(r)}]$.

3.3 Improvement of opinion retrieval using the MR model

In this section, we introduce two state-of-the-art unified lexicon-based opinion retrieval models, and then discuss how to improve these models using our proposed model.

Zhang and Ye [12] in their generation unified model (GUM) assumed that the latent variable s is a pre-constructed bag-of-words sentiment thesaurus, and all sentiment words s_i are uniformly distributed. It then ranks the document according to the probability that the document d contains relevant opinions to query q, which is given by

$$p(d|q, s) \propto \frac{1}{|s|} \sum_i p(s_i|d, q) p(q|d) p(d). \tag{11}$$

Obviously, it is unreasonable to assume that the sentiment words in GUM are uniformly distributed and weighted by just relying on the co-occurrence of the sentiment word and query word inside the document. To enhance the performance of the algorithm, we replace the weight of sentiment words s_i with our *MR model*.

Luo et al. [14] proposed an LTR model integrating social features and opinionated features in addition to traditional topic-relevant features. A corpus-derived lexicon was used to construct an opinion score for each tweet. In particular, the opinionate score of each tweet d is calculated by the average opinion over certain terms. The presented method uses the chi-squared value to estimate the opinion score of a term t. The estimated formula is as follows:

$$Opinion_{avg}(d) = \sum_{t \in d, \chi^2(t) \geqslant m} p(t|d) \cdot Opinion(t), \tag{12}$$

where $Opinion(t)$ is the opinion score of term t and defined by a function of the chi-squared value of a term $\chi^2(t)$.

However, the performance of the LTR model can be improved by computing $Opinion(t)$ with a new sentiment word weighting schema resulting from our MR model, which integrates both document information and social interaction information.

4 Experimental evaluation

4.1 Dataset

To the best of the authors' knowledge, there is no annotated microblog dataset with contextual social interactions of users to evaluate our proposed model. Therefore, we created a new real dataset from Sina Weibo, a typical Chinese microblogging platform that enables users to follow any other users and receive comments from those followed users. In Sina Weibo, users can publish brief posts, comment on other users, and spread information by forwarding the posts. We crawled posts, comments, and contextual social relationships by using Sina Weibo API from May 16, 2015 to June 6, 2015.

Tables 2 and 3 show the descriptions as well as some statistics of our dataset. The dataset is constructed from three domains: sports & electronics, movies & entertainment, and politics & news. Employing the popular pooling paradigm, three annotators were trained to label the dataset independently. Given a collection of the top 200 posts from each query, the annotation task is to manually label each post as subjective or objective with respect to the given topic. According to the majority voting heuristic, the final label of the post is assigned by the majority of three annotators.

In the sports & electronics domain, 57.6 out of 200 posts on average are labeled as subjective accounting for 28.8%. In the movies & entertainment domain, 65.2 out of 200 posts on average are labeled as subjective accounting for 32.6%. In the politics & news domain, 44.7 out of 200 posts on average are labeled as subjective accounting for 22.3%. Furthermore, the median numbers of comments per post on average in the electronics and sports, movie and entertainment, and politics and news domains are 40.6, 186.3, and 126.1, respectively. The median numbers of followers per user on average in the electronics and sports, movies and entertainment, and politics and news domains are 296.4, 396.3, and 622.3, respectively. As illustrated in Table 3, we use 30 topics to evaluate the comparative methods. The domain distributions of the topics are 7, 10, and 13 for the sports and electronics, movies and entertainment, and politics and news domains, respectively.

Table 2 Statistics of the dataset

Topic domain		Sports and electronics	Movies and entertainment	Politics and news
Posts per topic		200	200	200
Average opinionate posts per topic		57.6	65.2	44.7
Average users per topic		71.7	124.4	133.6
Average comments per post	Minimum	0	0.1	0
	Median	40.6	186.3	126.1
	Maxine	2309.4	3149.7	3599.4
Average followers per user	Minimum	15.1	4.1	1.4
	Median	296.4	395.3	622.3
	Maxine	10812.9	153158.3	21758.8

Table 3 The description of topics

Id	Topic	Id	Topic	Id	Topic
1	Jing Dong	11	Yongyuan Cui	21	Real Madrid
2	Civil servant	12	Barcelona	22	Kobe Bryant
3	Fleet of time	13	Hengda	23	Programmer
4	Jay Chou	14	Spring Festival Gala	24	James
5	Chinese Football Team	15	The Taking of Tiger Mountain	25	Benshan Zhao
6	Local tyrant	16	Charlie weekly	26	Transgenosis
7	City inspectors	17	One Piece	27	Ziqi Deng
8	Beina Yao	18	Lakers	28	He Chen
9	MI phone	19	Naruto	29	Haze
10	Loser	20	Sicong Wang	30	MaYing-jeou

4.2 Evaluation metrics

The metrics adopted for evaluating our proposed methods are mean average precision (MAP) and normalized discounted cumulative gain (NDCG).

The MAP metric is defined as

$$\mathrm{MAP} = \sum_{i=1}^{Nq} AP_i/Nq, \tag{13}$$

where AP_i is the average precision for the ith query and Nq is the total number of queries. The average precision for each query is as follows:

$$AP = \frac{1}{\sum_{i=1}^{N} r_i} \sum\nolimits_{i=1}^{N} r_i\left(\frac{\sum_{j=1}^{i} r_j}{i}\right), \tag{14}$$

where N is the total number of documents and $r_i = 1$ denotes ith document is opinionate document.

The NDCG is a measure of ranking quality, which is given by

$$\mathrm{NDCG}@n = Z_n \sum_{j=1}^{n} (2^{r(j)} - 1)/\log(1 + j), \tag{15}$$

where j is the document position in the result list and $r(j)$ presents the relevance of the document in position j.

4.3 Comparison with other methods

In our experiments, we adopted HowNet (see HowNet official websitc), which is widely applied to lexicon-based opinion detection methods and proved effective in many works. It consists of 836 positive sentiment words, 3,730 positive comments words, 1,254 negative sentiment words, and 3,116 negative comments words. To assess the performance of our proposed model, we compared our MR-based models with two typical lexicon-based opinion retrieval models on the real Sina Weibo dataset (described in Section 4.1).

- **GUM** is the method proposed in [12]. It is a novel probabilistic generation model that unifies the topic relevance score and opinion score. The retrieval performance under different external sentiment thesauruses, such as HowNet, WordNet, General Inquirer, and SentiWordNet, is compared and discussed empirically, and the cross-language HowNet dictionary performs better than all other candidates.
- **LTR** is the method described in [14]. It employs an LTR model for integrating social and opinionate information for tweets opinion retrieval. A corpus-derived lexicon is adopted to estimate the opinionated score of each tweet by calculating the average opinion score over certain terms. They used the chi-squared value, based on a manually tagged subjective tweets set and objective tweets set, to estimate the opinion score of a term.
- Our generation model (**GMR**) is the method proposed in this paper, which employs GUM's probabilistic generation model. Unlike GUM's method relying on sentiment words from an external sentiment thesaurus, our model uses the weights of sentiment words that are computed by our proposed MR model.

- Our LTR model (**LMR**) is the method presented in this paper. In this model, we adopt Luo et al.'s LTR framework, and estimate opinionate score of each microblog post based on our MR model. Furthermore, the resulting opinionate scores of a user's posts are treated as features for the LTR framework.

In reviewing these comparative models, GUM is a unified generation model for opinion retrieval ranking post only using post information, while LTR is a machine learning model integrating social and opinionate information. These two comparative methods estimate the strength of sentiment words from the knowledge database HowNet and Twitter corpus information, respectively, while our GMR and LMR approaches incorporate social contextual relationships to imply the strength of sentiment words selected from HowNet. The parameter settings of GUM and LTR are the same as in the original papers.

To reduce the variance of performance estimation, six-fold cross-validation is applied to evaluate the four comparative methods. As for the weight matrix W in our proposed MR model, we set it as

$$\begin{bmatrix} 1 & 0.5 & 0.5 & 0.25 \\ 0.5 & 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0 & 0.5 \\ 0.25 & 0.25 & 0.25 & 0 \end{bmatrix}$$

based on the hierarchical structure among user–post/comment–word. Here W is normalized to be column stochastic. The results of each comparative method on the real Sina Weibo dataset are shown in Table 4. From the table, we make the following observations.

- Our proposed methods (GMR and LMR) improve the performance of two other baselines significantly according to MAP and NDCG metrics. Our LMR method achieves the best performance in MAP and NDCG@5, while GMR method obtains the best performance in NDCG@10 and NDCG@15.
- As for the probabilistic generation model, our GMR method outperforms the GUM in both MAP and NDCG. Specifically, compared with the GUM, our GMR method obtains 19.8% improvement in MAP and achieves 73.4%, 88.8%, and 86.0% improvement in NDCG@5, NDCG@10, and NDCG@15, respectively.
- For the machine learning model, our LMR model is significantly better than the LTR model according to the MAP and NDCG metrics. It outperforms the LTR model by 14.0%, 14.5%, 10.1%, and 8.6% in MAP, NDCG@5, NDCG@10, and NDCG@15, respectively.
- Compared with the improvement of our LMR method on the LTR method, our GMR method achieves greater improvement on the GUM method. The reason for this phenomenon is that the GUM method ranks posts only relying on document information, while the LTR method scores posts according to social and opinionate information.

Table 4 Results of comparative approaches

Approach	MAP	NDCG@5	NDCG@10	NDCG@15
GUM	0.295	0.403	0.457	0.481
LTR	0.349	0.629	0.659	0.664
GMR	0.353	0.699	0.863	0.895
LMR	0.398	0.721	0.726	0.722

To study how the performance of MR-based methods is influenced by different queries, per MAP gain on individual topics by GMR and LMR are shown in Figs. 2 and 3.

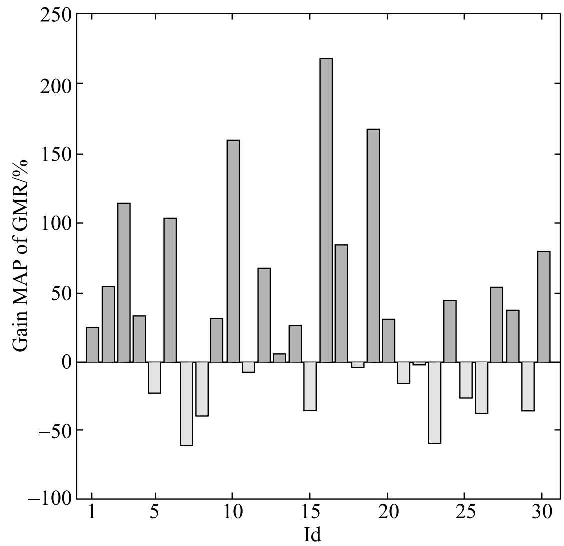


Fig. 2 Per-topic analysis: MAP improvement over 30 topics in the GMR model

As shown in Fig. 2, for most queries the GMR method outperforms the GUM method. Topics 16, 10, and 19 achieve 230%, 160%, and 165% MAP gains over the GUM method, respectively. However, topics 7 and 23 obtain −60% and −55% MAP gains over the GUM method.

As illustrated in Fig. 3, the MAP performance of the LMR method does not gain very significant improvement over the LTR method according to the numbers of queries. Topics 16 and 25 obtain 22% and 18% MAP gains over the LTR method, respectively, which is in the top two MAP gains. Except for these two topics, the LMR method does not perform very well in most of the others according to the MAP metric. The reason is that the LTR method employs Twitter-specific features and author features in the LTR framework as well as document information. However, the LMR model outperforms the LTR method and obtains more than 8.6% gain in NDCG@5, NDCG@10, and NDCG@15.

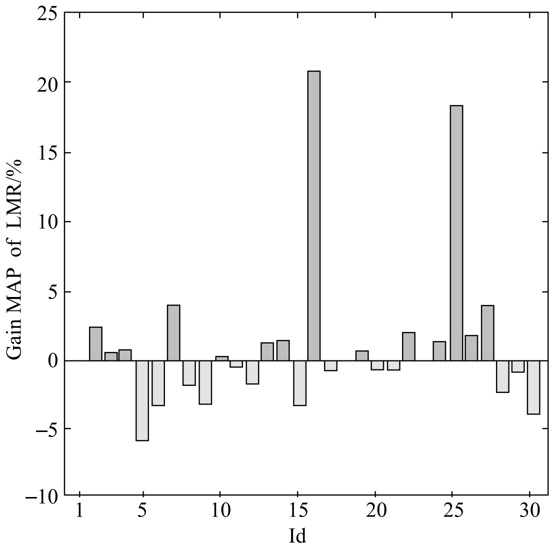


Fig. 3 Per-topic analysis: MAP improvement over 30 topics in the LMR model

4.4 Parameter tuning

Different settings of parameters may affect the quality of MR sentiment lexicon ranking. The aim of the following experiments is to examine the performance of the MR ranking model while varying the involved parameters. There are three types of parameters in our algorithm: (1) the weight parameters α_1–α_4 adopted for the authority of users; (2) the parameters β_1–β_4 used for the weight of posts and comments; (3) the parameters γ_1–γ_4 applied for the weight of sentiment words.

In these experiments, we initiate the weight parameters according to the structure relations between every two among the four levels: user, post, comment, and sentiment words by intuition. As for the user level, the initial parameters are set as $\alpha_1 = 1, \beta_1 = 0.5$, and $\gamma_1 = 0.25$. For the post level, we set the parameters as $\alpha_2 = 0.5, \beta_2 = 0.5$, and $\gamma_2 = 0.5$. For the comment level, the parameters are set as $\alpha_3 = 0.5$ and $\gamma_3 = 0.5$. For the sentiment word level, we set the parameters as $\alpha_4 = 0.25$ and $\beta_4 = 0.25$. Furthermore, our MR methods are run by adjusting successively the parameters of the four levels: user, post, comment, and sentiment words. Each time we tune each parameter ranging from 0.1 to 1.0 with a step size of 0.1, while the remaining parameters are fixed.

As for GMR model, the reinforcements among comment–user, post/comment, and user–post/comment interactions are considered more important than other interactions, which enhance the performance of our proposed MR model as illustrated in Fig. 4. Our GMR method achieves the best MAP in the user level while the parameters are set as $\alpha_1 = 0.4$, $\beta_1 = 0.5$, and $\gamma_1 = 0.1$. It obtains a 15.9% improvement over the GUM method. In the post level, the GMR model gains an increase of 16.1% in MAP over the GUM method when we set the parameters as $\alpha_2 = 0.3$, $\beta_2 = 0.5$, and $\gamma_2 = 0.1$. In the comment level, the GMR method obtains an increase of 19.8% in MAP over the GUM method when the parameters are set as $\alpha_3 = 0.8$ and $\gamma_3 = 0.2$. In the sentiment word level, the GMR method obtains an increase of 19.8% in MAP over the GUM method when the parameters are set as $\alpha_4 = 0.25$ and $\beta_4 = 0.25$. As illustrated in Figure 4, the parameter γ_3 has a major influence on the GMR model. The reason for this is that compared with other factors, the sentiment word is the most important factor for ranking the opinion score of short microblogging posts.

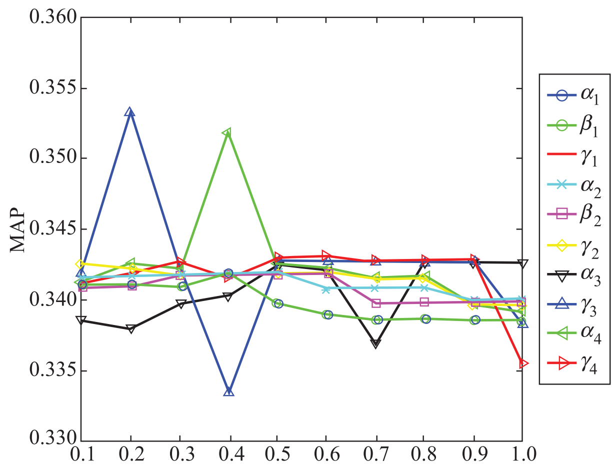


Fig. 4 Performance of MAP with varying parameters in the GMR model

Finally, we examine the performance of the LMR model by varying the involved parameters as shown in Fig. 5. It can be observed that the reinforcement of post–comment, sentiment–user, and user–user interactions is more significant than that of other interactions for the performance improvement of our MR model. In the user level, our LMR model achieves the best performance in MAP and obtains 13.6% improvement over the LTR model when the parameters are set as $\alpha_1 = 0.5, \beta_1 = 0.5$, and $\gamma_1 = 0.25$. In the post level, the LMR model achieves an increase of 13.9% in MAP over the LTR model when we set the parameters as $\alpha_2 = 0.1, \beta_2 = 0.9$, and $\gamma_2 = 0.5$. In the comment level, the LMR method obtains an increase of 13.9% in MAP over the LTR method when the parameters are set as $\alpha_3 = 0.5$ and $\gamma_3 = 0.5$. In the sentiment word level, the LMR method gains an increase of 14.1% in MAP over the LTR method when the parameters are set as $\alpha_4 = 0.9$ and $\beta_4 = 0.25$.

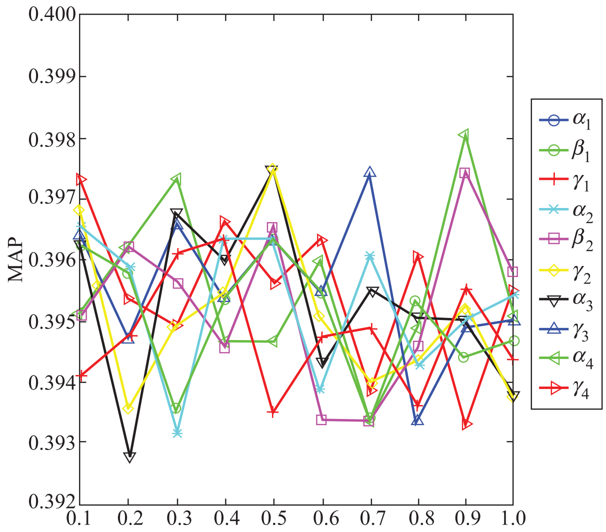


Fig. 5 Performance of MAP with varying parameters in the LMR model

5 Conclusions

In this paper, we have focused on the problem of Chinese microblog opinion retrieval. Different from previous approaches that rank each post relying on online public sentiment dictionaries or corpus-derived lexicons, we have proposed an MR model that integrates the microblog's social contextual information and document information to estimate the strength of sentiment words. To do that, the social contextual relations among user, post/comment, and sentiment words have been exploited to construct a unified three-layer heterogeneous graph. Then, a random walk sentiment word weighting algorithm was presented to capture more precise opinion strength of the sentiment words. Finally, the resulting weights of the sentiment word have been incorporated into the two state-of-the-art lexicon-based models (i.e., the GUM and LTR models) for Chinese microblog opinion retrieval. Experimental results on a real dataset have shown that our proposed model could achieve better performance than the comparative methods.

The novelty of our work lies in integrating user social contextual relations and document information to calculate the strength of sentiment words. Based on the "user–user" interpersonal interactions, "user–post/comment" intrapersonal interactions, and "post/comment–sentiment word" relations, a unified three-layer heterogeneous graph has been built. Our proposed random walk algorithm can benefit from estimating the strength of sentiment words by mining social contextual features and document features on the heterogeneous graph. As for future directions on opinion retrieval, more research should be focused on how to exploit the rich social media features for improving the performance of mining algorithms.

Acknowledgements This work was partially supported by the National Natural Science Foundation of China (Grant No. 61300105), the Research Fund for Doctoral Program of Higher Education of China (2012351410010), the Key Laboratory of Network Data Science & Technology, Chinese Science and Technology Foundation (CASNDST20140X), the Key Project of Science and Technology of Fujian (2013H6012), the Project of Science and Technology of Fuzhou (2012-G-113 and 2013-PT-45), and the Scientific Research Project of the Educational Department in Fujian Province (JA10055).

References

1. Pang B, Lee L. Opinion mining and sentiment analysis. Journal of Foundations and Trends in Information Retrieval, 2008, 2(1–2): 1–135
2. Macdonald C, Ounis I, Soboroff I. Overview of the TREC-2007 blog track. In: Proceedings of the 16th International Text Retrieval Conference. 2007, 31–43
3. Seki Y, Evans D K, Ku L W, Sun L, Chen H H, Kando N, Lin C Y. Overview of multilingual opinion analysis task at NTCIR-7. In: Proceedings of the 7th National Center for Science Information Systems Test Collections for IR Meeting on Evaluation of Information Access Technologies. 2008
4. Tan S B, Liu K, Wang S G, Yan X, Liao X W. Overview of Chinese opinion analysis evaluation 2013. In: Proceedings of the 5th Chinese Opinion Analysis Evaluation. 2013, 5–33
5. Li B Y, Zhou L J, Feng S, Wong K F. A unified graph model for sentence-based opinion retrieval. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. 2010, 1367–1375
6. Zhang W, Yu C, Meng W Y. Opinion retrieval from blogs. In: Proceedings of the 16th ACM Conference on Information and Knowledge Management. 2007, 831–840
7. Santos R L T, He B, Macdonald C, Ounis I. Integrating proximity to subjective sentences for blog opinion retrieval. In: Proceedings of the European Conference on Information Retrieval. 2009, 325–336
8. Jiang L, Yu M, Zhou M, Liu X H, Zhao T J. Target-dependent twitter sentiment classification. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics. 2011, 151–160
9. Wang X L, Wei F R, Liu X H, Zhou M, Zhang M. Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In: Proceedings of the 20th ACM International Conference on Information and Knowledge Management. 2011, 1031–1040
10. Hu X, Tang J L, Gao H J, Liu H. Unsupervised sentiment analysis with emotional signals. In: Proceedings of the 22nd International Conference on World Wide Web. 2013, 607–618
11. Eguchi K, Lavrenko V. Sentiment retrieval using generative models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2006, 345–354
12. Zhang M, Ye X Y. A generation model to unify topic relevance and lexicon-based sentiment for opinion retrieval. In: Proceedings of the

31st Annual International ACM Special Interest Group on Information Retrieval Conference on Research and Development in Information Retrieval. 2008, 411–418

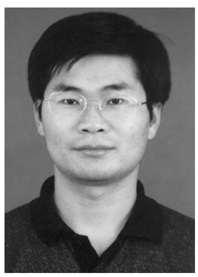
13. Huang X J, Croft W B. A unified relevance model for opinion retrieval. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. 2009, 947–956
14. Luo Z C, Osborne M, Wang T. Opinion retrieval in twitter. In: Proceedings of the International AAAI Conference on Weblogs and Social Media Conference. 2012, 507–510
15. Mei Q Z, Ling X, Wondra M, Su H, Zhai C X. Topic sentiment mixture: modeling facets and opinions in weblogs. In: Proceedings of the 16th International Conference on World Wide Web. 2007, 171–180
16. He B, Macdonald C, He J, Ounis, I. An effective statistical approach to blog post opinion retrieval. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. 2008, 1063–1072
17. Weerkamp W, Balog K, de Rijke M. A generative blog post retrieval model that uses query expansion based on external collections. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing. 2009, 1057–1065
18. Jiang M, Cui P, Liu R, Yang Q, Wang F, Zhu W W, Yang S Q. Social contextual recommendation. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management. 2012, 45–54
19. Feng S, Song K S, Wang D L, Yu G. A word-emoticon mutual reinforcement ranking model for building sentiment lexicon from massive collection of microblogs. World Wide Web Journal, 2015, 18(4): 949–967
20. Hu X, Tang L, Tang J L, Liu H. Exploiting social relations for sentiment analysis in microblogging. In: Proceedings of the 6th ACM International Conference on Web Search and Data Mining. 2013, 537–546
21. Li F T, Liu N N, Jin H W, Zhao K, Yang Q, Zhu X. Incorporating reviewer and product information for review rating prediction. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence. 2011, 1820–1825
22. Wei F R, Li W J, Lu Q, He Y X. Applying two-level reinforcement ranking in query-oriented multidocument summarization. Journal of the American Society for Information Science and Technology, 2009, 60(10): 2119–2131

Jingjing Wei is currently an assistant professor at Fujian Jiangxia University, China. She received her PhD degree from Fuzhou University, China in 2017. Her research interests include natural language processing, information retrieval and extraction, and social media analysis.

Xiangwen Liao is currently an associated professor at Fuzhou University, China. He received his PhD degree in computer architecture from Institute of Computer Sciences, Chinese Academy of Sciences, China in 2009. His research interests include natural language processing, information retrieval and extraction, and social media analysis.

Houdong Zheng received his Master degree in computer science from College of Mathematics and Computer Sciences, Fuzhou University, China in 2017. His research interests include natural language processing, information retrieval and extraction, and social media analysis.

Guolong Chen is currently a professor at Fuzhou University, China. He received his PhD degree in computer science from Xi’an Jiaotong University, China in 2002. His main research interests include computing intelligence, network science, big data processing, very large scale integration, etc.



Xueqi Cheng is a professor at Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), China. He received his PhD degree in computer science from ICT, CAS in 2006. His current research interests include social computing, information retrieval, big data analysis, etc.