





# An Interactivity-Based Personalized Mutual Reinforcement Model for Microblog Topic Summarization

Lu Zhang<sup>1,2</sup> , Liangjun Zang<sup>1</sup> , Longtao Huang<sup>1</sup>, Jizhong Han<sup>1</sup>,  
and Songlin Hu<sup>1,2</sup>

<sup>1</sup> Institute of Information Engineering, Chinese Academy of Sciences,  
Beijing 100093, China  
{zhanglu, zangliangjun, huanglongtao, hanjizhong,  
husonglin}@iie.ac.cn

<sup>2</sup> School of Cyber Security, University of Chinese Academy of Sciences,  
Beijing 100049, China

**Abstract.** Microblog topic summarization aims to provide readers with a concise and representative set of high-quality messages for efficient digestion. Unlike previous traditional text summarization, microblog posts are accompanied by user interactive behaviors such as thumbs-ups, comments and retweets. Intuitively, messages with more social influence are more important and should have higher probability to be included in the summary. We proposed a two-stage approach to solve microblog topic summarization problem, named Interactivity based Personalized Mutual Reinforcement Model (IPMR Model). Firstly, we extract keywords from all posts of a given topic and then collect relevant sentences which contain keywords as candidates. Secondly, we rank candidate sentences using a mutual reinforcement model based on interactivity statistics with keywords. We conducted a series of experiments on a corpus of microblog posts from five topics, and used Rouge-N as evaluation metric. Experimental results show that our model achieves the best overall performance consistently when compared with several state-of-the-art models.

**Keywords:** Microblog topic summarization · Social interactivity  
Keywords extraction · Mutual reinforcement model

## 1 Introduction

Microblogging is one of the most popular social entertainment applications in recent years. The most representative examples may be *Twitter* (English) and *Sina Weibo* (Chinese). Taking *Sina Weibo* for example, according to accounting data of December 2017, it had 392 million active users per month.<sup>1</sup> People use microblog platform to interact with their favorite stars, share happiness of their lives and so on. *Microblog Topic* is the most popular item among various services, where users express and share

---

<sup>1</sup> See <http://tech.sina.com.cn/i/2018-02-13/doc-ifyrmfmc2341675.shtml>.

their views on a topic by adding the corresponding hashtag to posts. Each topic has a separate page clustering posts with the same hashtag.

However, as there are many blog users and posting is convenient, the number of posts under hot topics is usually huge and posts quality is low. Thus a nice summary is needed to provide readers with representative messages for efficient digestion. To solve this problem, Chinese *Sina Weibo* has set up the ‘*elite posts*’ page under each topic managed by human hosts. However the selection and maintenance of ‘*elite posts*’ is a time-consuming job and may be subjective, which motivate researchers to study various methods to rank and select salient and diversified messages as a summary of each topic.

Recently researchers began to study on microblog topic summarization in Chinese. But it has many difficulties which are different from tweets, such as lack of public available large-scale datasets and so on. A notable work is CMiner [1] proposed in 2016, which aims at topic opinion targets extraction and summarization for Chinese microblog topics. However, it does not consider the ‘*interactivity*’ characteristic of microblog messages, indicated by thumbs-ups, comments, and retweets. These attributes are also influential factors in determining whether a post or a sentence should be included in a summary.

We argue that a good summary should contain important sentences that satisfy the following conditions. Firstly, important sentences should be representative and central, i.e. being similar in content to many other sentences from the same topic. Secondly, important sentences should have more social influence, i.e. having more thumbs-ups, comments and retweets.

Our model is based on the Mutual Reinforcement Model [1, 2] by developing it with personalized ‘*interactivity*’ attribute of each microblog message, so we call it *Interactivity-based Personalized Mutual Reinforcement (IPMR) Model*. Summaries are generated in two steps. Firstly, keywords are extracted from posts under a certain topic, and all sentences which contain at least one keyword are selected as candidates. Secondly, all candidate sentences and keywords are ranked simultaneously according to the IPMR model, where we assign the initial weight and jump probability of each sentence based on its interactivity. Finally, we choose top sentences to form a summary. We evaluate automatically generated summaries by Rouge-N metric against the ‘*elite posts*’ selected by human hosts of each topic. Compared with four other state-of-the-art models, experimental results show that our model performs best.

Our contributions are listed as follows:

- (1) To the best of our knowledge, there is rare work about integrating ‘*interactivity*’ information into the task of microblog topic summarization.
- (2) We propose an Interactivity-based Personalized Mutual Reinforcement model for Chinese microblog topic summarization, which does not require any manually labeled data.
- (3) We show experimentally that the interactivity information of messages improves the quality of summaries greatly.

## 2 Related Work

### 2.1 Traditional Text Summarization

Researches of automatic text summarization have developed over fifty years with a mature system. These methods are divided into three types, extractive summarization, compressed summarization and abstractive summarization. Extractive summarization focuses on extracting salient sentences from original documents to form a summary. Abstractive summarization tends to generate each word automatically after training process. Compressed summarization is a compromise between the above two.

Early summarization generation methods are almost extractive, therefore extractive summarization system is the most complete one and its experimental results are also the best. Early algorithms select sentences based on some certain features by using unsupervised and supervised methods, called as feature-based methods. These features contain content word frequency [3], sentence position, named entities and so on. The task of selecting important sentences can also be seen as a classification problem, therefore many machine learning methods are utilized subsequently. For example, Markov models [4], conditional random fields [5] and so on.

Afterwards, some graph-based methods were proposed. The underlying assumption is that central sentences are more important as they carrying much information. Researchers defined central sentences as those which are more similar to other sentences. Typical ones are the LexRank algorithm [6] and the TextRank algorithm [7]. During this period, many other graph-based methods had been investigated. Zha [8] used mutual reinforcement principle to extract keyphrase and generate summaries simultaneously. Its theory is similar to the HITS algorithm [9] with a transition matrix between sentences and words. Wan et al. [2] developed [8] by adding the matrix between sentences and the other matrix between words in the same task. Its theory seems like the combination of PageRank [10] and HITS algorithm.

With the rapid growth of neural networks, there are more and more researches on abstractive summarization. In addition, abstractive summarization is more similar to the way that people writing summaries. Rush et al. [11] applied Convolutional Neural Network to the task of sentence-level summarization with a Neural Attention Model. Nallapati et al. [12] used sequence-to-sequence Recurrent Neural Networks model to address critical problems in summarization. However, it should be noted that abstractive summarization mainly focuses on headline generation of short texts. As the task is more difficult and neural network has just developed for a short time, there is still much room for improvement, such as readability, time complexity and so on. There is a survey of recent advances in document summarization from 2012 to 2017 by Yao et al. [13], besides latest approaches, it also forecasts the possible directions of future development.

### 2.2 Microblog Summarization

In recent years, the field of text summarization has changed dramatically. With the advent of the WEB 2.0 era, the need for summaries has gradually shifted to cyberspace texts. For example, summarization for products reviews on some websites [14],

summarization for the Weibo blog posts and so on. When solving these new problems, extractive summarization methods are widely used.

Weng et al. [15] integrated posts and responses into the training model. It is a two-step scheme, classifying posts with responses into five classes, then applying different strategies to generate summaries for different classes, such as opinion analysis, response relevancy and so on. Bora [16] proposed a sentiment classification system for tweets. He built a Naïve Bayes Classifier to identify positive or negative emotions in posts, then generate public opinions summarization. Sharifi et al. [17] proposed the phrase reinforcement algorithm to generate summaries automatically. It aims at finding the most commonly used phrase as the summary. It usually begins with a trending topic as the root node, and adjacent of it are chains of common sequences of words collected from input posts. Zhao et al. [18] applied event detection to generate microblogs summarization by HITS algorithm. TREC 2017 has a relevant task called Real-Time Summarization Track [19], it aims at providing social media users with ‘interest posts’ timely. This task involves ‘interest profiles’ problem.

There is little work on the Chinese microblog summarization. Zhou et al. [1] presents a system *CMiner* that extracts opinion targets and opinion sentences as summaries simultaneously. They use a co-ranking algorithm in the model which is based on mutual reinforcement theory. Our task is different from it, since we devote to utilize posts specific ‘interactivity’ attribute to select sentences and our main task is generating summaries. Duan and Chen [20] designed a model of ranking tweets using social influence and content quality also in a mutually reinforcing manner. However they use user information such as authority to rank posts, in contrast, we pay more attention to the nature of posts. And these information is hard to acquire completely from Weibo, therefore we use only interactive statistics of posts like thumb-ups.

### 3 Our Model

In this section, we give a synthetic explanation for our IPMR Model. At first, we formulate the problem of microblog topic summarization as follows: given a topic  $t$  of *Sin-a Weibo*, we collect blogs of the topic. The collection can be regarded as a document, and we extract important sentences to form a summary. As microblog texts have specific attributes, the requirements for good summaries are different from traditional texts. We define ‘important sentences’ from the following two aspects:

- a. Content. It means that important sentences should have high centrality in content. This feature is widely used in traditional extractive text summarization.
- b. Interactivity. Microblog text has a specific property, interactivity. It is helpful for us to construct a summary containing hot opinions that interest people. Blog’s interactivity can be measured by its retweets numbers, comments numbers and thumbs-up numbers.

There are two simple observations: (1) *elite posts* usually contain topical keywords; (2) *elite posts* usually have high interactivity such as large retweets. Therefore, we extend the mutual reinforcement model [1] which performed well on blogs summarization with two new factors: topical keywords and interactivity statistics.

Our model can be divided into two parts. In the first step, we extract keywords from blogs collection. Then we collect sentences that containing at least one keyword as candidates. In the second step, we applies the personalized mutual reinforcement model combining the keywords set and the candidate sentences set. The underlying principle of this model is Personalized PageRank algorithm by using interactivity to personalize sentences and HITS algorithm. In next sections, we will demonstrate these two main steps in detail. Figure 1 presents the structure of our model.

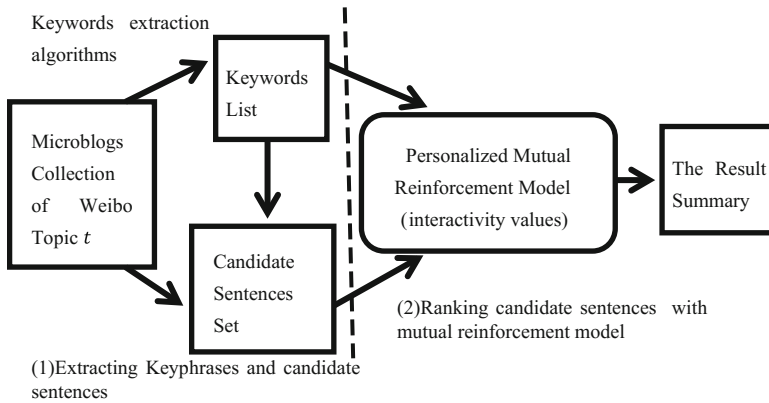


Fig. 1. Experimental method structure diagram

### 3.1 Extracting Keywords and Candidate Sentences

Based on the observation that elite posts usually contain topic central words which are similar to the definition of keywords, we integrated keywords into reinforcement model. As ideal summaries should have high coverage rate, we suppose that the number of keywords is the same as the number of clusters of blogs opinion targets. Zhou [1] proposed the method of obtaining the clusters number  $C$  based on *affinity propagation*. We compared typical keywords extraction methods by integrating them into our model, such as TF-IDF [21] and TextRank [22]. According to the experimental result, TextRank is more suitable for our model. After getting the keywords list, we collect relevant sentences that containing at least one keyword forming the candidate sentences set.

### 3.2 Ranking Candidate Sentences Based on Personalized Mutual Reinforcement Model

Keywords and candidate sentences sets are inputs of personalized mutual reinforcement model, and its output is the sorting results of these two datasets. Afterwards, top- $C$  (the size of keywords list, in Sect. 3.1) sentences are selected to form the result summary. To ensure that popular sentences are selected, we developed the general reinforcement model by personalizing it with interactivity values. The other underlying assumption is

the co-occurrence of salient sentences and keywords, the similarities of keywords and the similarities of salient sentences.

**Interactivity.** Before introducing the second model, we should give a clear explanation for posts ‘interactivity’. Interactivity is the most prominent attribute of microblog texts. On Sina Weibo platform, users can retweet, comment and thumb up others’ posts and these three operations reflect their attitudes. Retweeting represents users tend to share the post with others, commenting means users are interested in the blog, and thumbs-up stands for a feeling of approval. Therefore, the number of these three operations is closely related to the interactivity of microblogs text. In order to generate the ideal Weibo topic summarization, we should take these three values as measures of interactivity into consideration.

We assumed that all sentences in a blog share the same interactivity of this blog. We formulated the interactivity as a sum of linear addition of above three values, and calculated it as follows:

$$Score_{interactivity} = a \cdot x + b \cdot y + c \cdot z \quad (1)$$

Where  $a, b, c$  stand for three normalized coefficients. Their initial values are set according to their correlation coefficients to ‘elite posts’ (see more information in Sect. 5.1). And  $x, y, z$  are set to the log value of the number of retweets, comments and thumbs-up respectively.

**Personalized Mutual Reinforcement Model.** Our model based on interactivity aims at increasing the probability of hot sentences been selected. In our mutual reinforcement model, there is a mutually reinforcing relationship between keywords and candidate sentences. On one hand, a sentence (word) which is similar to more other sentences is assumed to be more representative. On the other hand, a sentence (word) which is linked to more salient words (sentences) is also assumed to be more important. Following the PageRank and the HITS paradigm, we formulate above two types of relationship as four random walks, one among words, one among candidate sentences, one from keywords to sentences and one from sentences to keywords. It should be noted that initial weights and jump probabilities of sentences are assigned based on interactivity. The intuitive interpretation is that sentences with high interactivity are more important (personalizing initial weights) and have more influence on others (personalizing jump probabilities).

For keywords set  $A$ , we define the transition matrix  $K \in \mathbb{R}^{|A| \times |A|}$ . Value  $K_{ij}$  represents similarity between word  $A_i$  and  $A_j$ , we calculate it based on Jaccard Index:

$$K_{ij} = \frac{|Character(A_i) \cap Character(A_j)|}{|Character(A_i) \cup Character(A_j)|} \quad (1 \leq i, j \leq |A|) \quad (2)$$

The function  $Character(A_x)$  is the Chinese character set of the  $x$  th word in set  $A$ , and  $|A|$  is the total number of words in set  $A$ . Then we normalize the matrix and make the sum of each row is 1.

$$K_{ij} = \frac{K_{ij}}{\sum_{k=1}^{|A|} K_{ik}} \tag{3}$$

In the rand walk among words which are regarded as vertexes, there are two kinds of jump in each step, one is jumping to connected words and the other is jumping randomly to any words in the graph with probability  $p$ . This update iteration process can be formulated as:

$$K_{t+1} = (1 - p)K_t + p \cdot M_k \tag{4}$$

Where  $M_k \in \mathbb{R}^{|A| \times |A|}$  is a matrix with all the values equal to 1.

For candidate sentences set  $B$ , we define the transition matrix  $S$  accordingly:

$$S_{ij} = \frac{\text{CosineSimliarity}(B_i, B_j)}{\sum_{k=1}^{|S|} \text{CosineSimliarity}(B_i, B_k)} \quad (1 \leq i, j \leq |B|) \tag{5}$$

Where  $\text{CosineSimliarity}(B_i, B_j)$  refers to calculating the similarity of sentence vectors of  $B_i$  and  $B_j$ ,  $|B|$  is the total number of sentences in set  $B$ .

The random walk model among sentences is similar to the one among words, and the result matrix of each iteration is formulated as:

$$S_{t+1} = (1 - p)S_t + p \cdot M_s \tag{6}$$

It should be noted that the matrix  $M_s \in \mathbb{R}^{|B| \times |B|}$  is different from  $M_k \in \mathbb{R}^{|A| \times |A|}$ . Its values are assigned by sentences interactivity scores as follows:

$$M_s \text{ ij} = \frac{\text{inter}(B_j)}{\sum_{j=1}^{|B|} \text{inter}(B_j)} \quad (1 \leq i, j \leq |B|) \tag{7}$$

Where  $\text{inter}(B_i)$  stands for the interactivity value of  $i$  th sentence.

To formulate the third and fourth random walks between keywords and candidate sentences, we define two matrices  $KS \in \mathbb{R}^{|A| \times |B|}$  and  $SK \in \mathbb{R}^{|B| \times |A|}$ .  $KS$  is the transition matrix from keywords to candidate sentences,  $SK$  is the transition matrix from candidate sentences to keywords. The value of these two matrices represents whether a word is included in a sentence, assigned as follows:

$$KS_{ij} = \frac{a_{ij}}{\sum_{k=1}^{|B|} a_{ik}} \quad (a_{ij} = 1 \text{ if } K_i \text{ in } S_j, \text{ else } a_{ij} = 0) \tag{8}$$

$$SK_{ij} = \frac{b_{ij}}{\sum_{k=1}^{|A|} b_{ik}} \quad (b_{ij} = 1 \text{ if } K_j \text{ in } S_i, \text{ else } b_{ij} = 0) \tag{9}$$

The mutual reinforcement model is a combination of these above four rand walks. As for a sentence (word), it can choose to jump to a sentence (word) of the graph at the

probability of  $\alpha$  (it is usually assigned as 0.85), or choose to jump to a word (sentence) at the probability of  $(1 - \alpha)$ . The initial weight of each word is assigned equally as  $1/|K|$ . As for candidate sentences, they are initialized according to its own interactivity. We defined two vectors for keywords and sentences separately:

$$\vec{k} = \left( \frac{1}{|A|}, \frac{1}{|A|}, \dots, \frac{1}{|A|} \right) \quad (k \in \mathbf{R}^{1*|A|}) \quad (10)$$

$$\vec{s} = \left( \frac{\text{inter}(B_1)}{\sum_{i=1}^{|B|} \text{inter}(B_i)}, \frac{\text{inter}(B_2)}{\sum_{i=1}^{|B|} \text{inter}(B_i)}, \dots, \frac{\text{inter}(B_{|B|})}{\sum_{i=1}^{|B|} \text{inter}(B_i)} \right) \quad (s \in \mathbf{R}^{1*|B|}) \quad (11)$$

Where  $\vec{k}$  stands for keywords vector,  $\vec{s}$  stands for sentences vector. In each step, they are updated as follows:

$$k_{t+1} = \alpha \cdot k_t \cdot K + (1 - \alpha) \cdot k_t \cdot KS \quad (12)$$

$$s_{t+1} = \alpha \cdot s_t \cdot S + (1 - \alpha) \cdot s_t \cdot SK \quad (13)$$

After a period of iterations, the result converges gradually. In the last step, we chose top- $C$  (the size of keywords list, in Sect. 3.1) sentences (deleting same sentences) according to the final values in the vector  $s$  as the result summary.

## 4 Experimental Settings

### 4.1 Datasets

We filtered tendentious topics related to stars and entertainment variety shows and compiled five datasets under five latest topics from December 2017 to February 2018, No. 1 ‘Jiang Ge trial’ (#江歌案庭审#), No. 2 ‘Couples Joint Debt’ (#夫妻共同债务#), No. 3 ‘the traveling frog game is brushing screen’ (#旅行青蛙刷屏#), No. 4 ‘the second verdict of the admonishing smoking causes the sudden death in an elevator trail’ (#电梯劝烟猝死案二审宣判#), No. 5 ‘hot search list and other items is temporarily closed for rectification’ (#热搜榜等版块暂时下线整改#). These topics involved different fields, and having different sizes. Therefore, they help us test the robustness of our model.

For each topic, we collected all posts it contained, collected contents and interactivity values (number of retweets, number of comments, and number of thumbs-up). After that, we did sentences segmentation for each blog as pre-processing. The assumption is that sentences in a microblog share the same interactivity of this blog. Posts in the ‘*elite posts*’ page under each topic are collected as standard datasets. Table 1 shows the detail information of each dataset.



**Table 1.** Dataset description (# denotes ‘number of’)

Topic number	#microblogs	#sentences	#‘elite posts’	Fields
1	108	<b>411</b>	<b>43</b>	International cases
2	39	<b>111</b>	<b>9</b>	Law
3	148	<b>259</b>	<b>42</b>	Game
4	37	<b>132</b>	<b>15</b>	Domestic cases
5	202	<b>292</b>	<b>10</b>	Regulations

## 4.2 Evaluation Metric

We use Rouge-N method to analyze results. It is the most popular evaluation method in texts summarization, and is based on computing n-gram recall rate. In experiments, we evaluate results based on  $N = 1, 2, 3, 4, 5, 6$ .

## 4.3 Baselines

We use four state-of-the-art methods as baselines. They are Cminer, extended-CMiner (adding interactivity), LexRank and Submodular.

CMiner: It was proposed in 2016, it aims to extract opinion targets and generate opinion summarization at the same time. It is applied to Weibo topic summarization and has good performance.

Extended-CMiner (adding interactivity): Based on the CMiner system, we developed it by adding the interactivity attribute as a new one. By comparing it with the original one, we can learn the influence of the interactivity attribute. Besides, by comparing our method with this model, we can analyze the influence of keywords on summarization.

LexRank and Submodular: LexRank is a graph-based model and Submodular performs well in deleting redundant information. Both of them are typical algorithms of extractive summarization. Therefore, we choose them as baselines to make experiments results more compelling.

# 5 Experimental Results

We designed a series of experiments, their evaluation results and analysis are shown in following sections.

## 5.1 Testing the Relevance Between Interactivity and ‘Elite Posts’

We conducted a correlation test on the Weibo topic # Jiang Ge trial #. There were 133 blogs and 516 sentences in this topic. After deleting the long reportable posts and irrelevant posts, there were altogether 108 microblogs with a total of 411 sentences. Collecting four attributes of posts: number of retweets, number of comments, number of thumbs-up, and whether it was an elite post. We performed logarithm operation on the first three attributes values. And if the post was an elite post, the fourth attribute

**Table 2.** T-tests and linear regression analysis results of three attributes

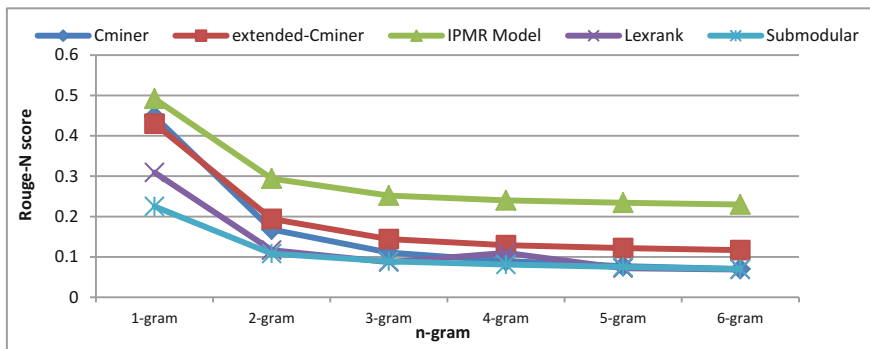
Operations	t	p	Corr-coef	Statistical conclusion
Retweet	2.222	0.028	0.191	significant ( $\alpha = 0.05$ )
Comment	2.086	0.039	0.180	significant ( $\alpha = 0.05$ )
Thumb up	3.127	0.002	0.227	extremely significant. ( $\alpha = 0.01$ )

would be set to 1 or 0 otherwise. We performed t-tests and linear regression analysis on the first three interactivity values with the fourth ‘is’ attribute respectively. The results are shown in Table 2.

According to Table 2, we can find that the correlation between posts’ interactivity and the judgement of ‘*elite posts*’. Among these three values, the number of thumbs-up has the greatest influence on the judgement of ‘*elite posts*’. Therefore, these three values as interactivity attribute can help us to pick out ‘*elite posts*’ which means that it is also useful for us to find salient sentences and generate a nice summary. And also we can get three correlation coefficients  $a$ ,  $b$ ,  $c$  for three operations correspondingly in the fourth column, which are used in the first equation (in Sect. 3.2).

## 5.2 Comparison with Baselines

We compare our IPMR Model with four baselines. We evaluate results by Rouge-N ( $N = 1, 2, 3, 4, 5, 6$ ) and compute the average of five datasets as final results. According to the data showed in Fig. 2, our model performs best and exceeds the second one a lot.

**Fig. 2.** Rouge-N experimental results

Firstly, We can see from Fig. 2 that two traditional baselines LexRank and Submodular have the lowest scores, and other three models are much better than them especially in 1-gram and 2-gram.

Then we compare CMiner and extended-CMiner, and the only difference between them is whether using ‘interactivity’ statistics. We can find that extended-CMiner

performs better than CMiner, hence the ‘interactivity’ factor is helpful to improve experimental results.

Finally, we compare our model with extended-CMiner to learn the influence of topical keywords. Their only difference is the first step, extended-CMiner extracts nominal phrase while our model extracts keywords. According to results in Fig. 2, we find that our model performs much better than extended-CMiner under any evaluation metric. Therefore, it is reasonable to conclude that generating summaries by using keywords can improve results significantly.

In conclusion, our model has the best overall performance.

### 5.3 Parameter Sensitivity Analysis

In this section, we study the sensitivity of the parameter  $p$  in the second part of our model, the personalized mutual reinforcement model. It determines the jump probability in the first two random-walks (among words, among sentences), referring to Eqs. (4) and (6). To test its sensitivity, we select 0.2, 0.4, 0.6, and 0.8 for four values. The experimental Rouge-2 results are shown in Fig. 3. Based on the result, we conclude that the influence of the parameter  $p$  is slightly and our model is robust. According to the graph, parameter  $p = 0.2$  has the best recall rate, we set it as 0.2 in experiments.

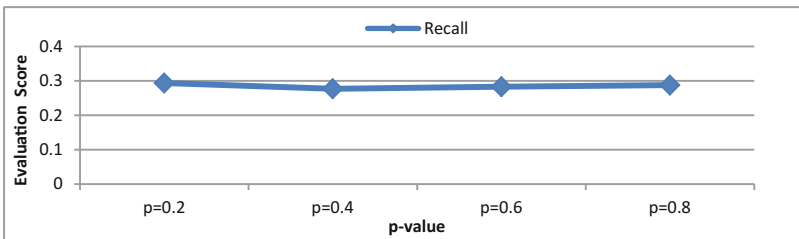


Fig. 3. Parameter  $p$  sensitivity analysis results

## 6 Conclusion and Future Work

In this paper we propose IMPR Model for automatic selection of important posts sentences as *Weibo* topics summaries, which consider interactive property of messages and topical keywords simultaneously. We show that the judgement of ‘elite posts’ is highly relevant to interactive property of messages. Specifically, we verify the importance of interactivity and show that using keywords in personalized mutual reinforcement model can greatly improve the quality of summary compared with nominal phrases. We compare our model with four other state-of-the-art methods using Rouge-N evaluation metric, and the result show that our model has the best overall performance.

In the future, we consider incorporating user information into the model, based on the assumption that posts of high-level users or users with a large number fans are usually in high quality. In this way, their posts are more conducive to guiding the public and creating a friendly cyberspace environment.

**Acknowledgments.** This work is supported in part by the National Key Research and Development Program of China (No. 2017YFB1010000) and the National Natural Science Foundation of China (No. 61702500).

## References

1. Zhou, X., Wan, X., Xiao, J., et al.: CMiner: opinion extraction and summarization for Chinese microblogs. *IEEE Trans. Knowl. Data Eng.* **28**(7), 1650–1663 (2016)
2. Wan, X., Yang, J., Xiao, J., et al.: Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In: *Proceedings of the 45th ACL*, pp. 552–559 (2007)
3. Vanderwende, L., Suzuki, H., Brockett, C., et al.: Beyond SumBasic: task-focused summarization with sentence simplification and lexical expansion. *Inf. Process. Manage.* **43**(6), 1606–1618 (2007)
4. Conroy, J.M., O'leary, D.P.: Text summarization via hidden Markov models. In: *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 406–407 (2001)
5. Shen, D., Sun, J.T., Li, H., et al.: Document summarization using conditional random fields. In: *International Joint Conference on Artificial Intelligence*, pp. 2862–2867 (2007)
6. Erkan, G., Radev, D.R.: LexRank: graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.* **22**(1), 457–479 (2004)
7. Mihalcea, R., Tarau, P.: TextRank: Bringing order into texts. In: Lin, D., Wu, D. (eds.) *Proceedings of EMNLP 2004*, Association for Computational Linguistics, pp. 404–411, Barcelona, Spain (2004)
8. Zha, H.: Generic summarization and key phrase extraction using mutual reinforcement principle and sentence clustering. In: *Proceedings of the 25th ACM SIGIR*, pp. 113–120 (2002)
9. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. *J. ACM* **46**(5), 604–632 (1999)
10. Page, L., Brin, S., Motwani, R., et al.: The PageRank citation ranking: bringing order to the web. Technical report. Stanford University, Stanford, CA (1999)
11. Rush, A.M., Chopra, S., Weston, J., et al.: A neural attention model for abstractive sentence summarization. In: *Empirical Methods in Natural Language Processing*, pp. 379–389 (2015)
12. Nallapati, R., Zhou, B., Santos, C.N., et al.: Abstractive text summarization using sequence-to-sequence RNNs and beyond. In: *Conference on Computational Natural Language Learning*, pp. 280–290 (2016)
13. Yao, J., Wan, X., Xiao, J.: Recent advances in document summarization. *Knowl. Inf. Syst.* **53**, 297–336 (2017). <https://doi.org/10.1007/s10115-017-1042-4>
14. Kim, S., Hovy, E.H.: Automatic identification of pro and con reasons in online reviews. In: *Proceedings of the ACL*, pp. 483–490, Barcelona, Spain (2006)
15. Weng, J., Yang, C., Chen, B., et al.: IMASS: an intelligent microblog analysis and summarization system. In: *Proceedings of ACL*, pp. 133–138, Barcelona, Spain (2011)

16. Bora, N.N.: Summarizing public opinions in tweets. *Int. J. Comput. Linguist. Appl.* **3**(1), 41–55 (2012)
17. Sharifi, B., Hutton, M., Kalita, J.K., et al.: Summarizing microblogs automatically. In: *Proceedings of North American Chapter of the Association for Computational Linguistics*, pp. 685–688 (2010)
18. Zhao, J., Liu, S., Liu, Y.: Towards event detection and summarization on microblogging platforms. *Adv. Sci. Technol. Lett.* **123**, 206–210 (2016)
19. TREC Real-Time Summarization Track Homepage. <http://treerts.github.io/>. Accessed 29 Mar 2018
20. Duan, Y., Chen, Z., Wei, F., et al.: Twitter topic summarization by ranking tweets using social influence and content quality. In: *Proceedings of the International Conference on Computational Linguistics*, pp. 763–780 (2012)
21. Mihalcea, R.: Graph-based ranking algorithms for sentence extraction, applied to text summarization. In: *Proceedings of 42nd ACL, Barcelona, Spain* (2004)
22. Zhou, D., Orshanskiy, S.A., Zha, H., et al.: Co-ranking authors and documents in a heterogeneous network. In: *Proceedings of International Conference on Data Mining*, pp. 739–744 (2007)