Using Queried Keywords or Full-text Extracted Keywords in Blog Mining?

Y. H. Chen, Eric J.- L. Lu *, and M. F. Tsai

***Abstract*—In recent years, increasingly worthy content and reliable information have been disseminated on blogs. This means that readers are accustomed to searching information from blogs. However, the number of blogs has increased tremendously, so it is difficult to identify related information in the enormous number of existing articles. In general, the keyword is used as the basic method for readers to filter unimportant information via search engines. Keywords can be seen as reflecting the essence of an article. That is, a full-length article can be referenced through several keywords. As a result, if the keywords chosen for an article are accurate, the users' intentions will be satisfied. Therefore, the methods for selecting accurate keywords are critical. One general approach generates keywords by applying full-text keywords retrieval process, but it is a time-consuming process for handling the enormous number of articles. In this paper, to save time in generating keywords via full-text keywords retrieval process, a system called Blog Connect is proposed to embed the tracing code for selecting queried keywords as keyword candidates. Experiments provided positive data to confirm the effectiveness of the proposed method.**

***Index Terms*—Blog network, blog mining, information retrieval**

I. Introduction

The blog is a major platform widely used on the Internet to share ideas, news, and entertainment [5][9]; a survey of Blogpulse [13] showed that the number of identified blogs is more than 150 million so far. The importance of blogs can be found in the survey from Technorati's 2010 report [13], which said that 40 percent of blog readers agree with the bloggers' opinions more than they trust mainstream media. Also, 48 percent of bloggers believe that net surfers will be receiving more of their ideas, news, and entertainment from blogs in the next five years than from traditional media. Despite the importance of blogs in information sharing, each blog is still considered an isolated island. That is, no connection or relationship between any two blogs is assumed, unless it was manually created by using blog-rolls, citation links, or comments [1][5]. If all related blogs were somehow "auto-magically" connected, it might result in a breakthrough in information sharing.

To establish relationships between any two blogs, the general approach is to calculate their similarity. The calculation can be based on either blog's tags/categories [18] or its content [4][5][12]. Bloggers often use tags or categories to classify their blog articles. However, this may result in the synonym problem that one same tag or category name is used to label an article but with different meanings. For example, the term "Java" has two meanings; one refers to a programming language and the other refers to coffee. For the calculation based on blog content, a full-text keyword retrieval process is generally required. The full-text keyword retrieval process consists of the following steps: download an article from remote site, scan the full-length article, segmentation, and summarize the number of occurrences for each keyword. This process is a complicated and time-consuming process. Furthermore, the contents of blog articles in reality may change which makes the process has to be done repeatedly.

It is a common practice that users enter keywords on search engines to find articles she need. Because it is believed that queried keywords represent users' intension [7][9][14], it will be interesting to find out whether or not the intension represents the topic of an article. If so, the time-consuming full-text keyword retrieval process can be eliminated. To verify this, we developed a platform called Blog Connect (BC) and used the tracing code embedded in the BC widget [2] to collect users' queried keywords. In this paper, we defined m-ratio which is the ratio of the number of matched queried keywords over the total number of queried keywords. The experiment yielded positive results to confirm that queried keywords can be used in the applications of blog mining. Also, the cost of using queried keywords is much lower than using full-text extracted keywords.

This paper is organized as follows: related works were briefly described in section 2. In section 3, BC and the process of collecting keywords were presented. The m-ratio and related experiments were illustrated in section 4. Finally, possible future works were concluded in section 5.

Manuscript received Dec. 30, 2011; revised January 16, 2012. This work was supported in part by the National Science Council, Taiwan, ROC, under contract no.: NSC100-2221-E-005-063.

Yi Hui Chen is with the Department of Applied Informatics and Multimedia, Asia University, Taichung, Taiwan (e-mail: chenyh@asia.edu.tw).

*Eric Jui-Lin. Lu. is with Department of Management Information Systems, National Chang Hsing University, Taichung, Taiwan, phone: 886-4-22840864#696; fax: 886-4-22857173; e-mail: jllu@dragon.nchu.edu.tw).

Meng Fang Tsai is with Department of Management Information Systems, National Chang Hsing University, Taichung, Taiwan. (e-mail: alisatime@gmail.com).

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

II. Related Work

The blog is a platform for information discovery and sharing. Some researchers [1][5][8] and commercial services [3][15] analyzed existing blogsphere in attempt to discover new or potential blogsphere.

Lu and Zhu[8] analyzed existing linkages, such as blogroll link, citation link, and comments, of blog articles to discover potential social network groups. Gao and Lai[5] employed formal concepts analysis (FCA) to cluster blogs based on full-length article content. Zhang et al. [18] proposed to use blog tags or categories to cluster blog articles. However, Hope et al.[6] argued that tag or category names may suffer from the synonym problem. Therefore, they proposed a semi-automatic tagging mechanism to provide bloggers' better tag names.

Other researches [10][12], namely, blog mining, applied information retrieval techniques to mine the information behind blog articles. Blog mining generally extracts keywords and selects major keywords, calculated based on the concept of term frequency (TF) or term frequency*inverse document frequency (TF*IDF), to represent the main topic of each article.

III. Our Approach

The proposed system, namely Blog Connect (BC) [2], is a cross-platform system developed to help bloggers to analyze incoming flows and provide means to create relationships among blogs with similar interests or topics. Registered bloggers can easily obtain a piece of tracing code which is written in Javascript and can be embedded in a blog widget called BC widget. The flow chart for collecting keywords from the tracing code is shown in Figure 1.

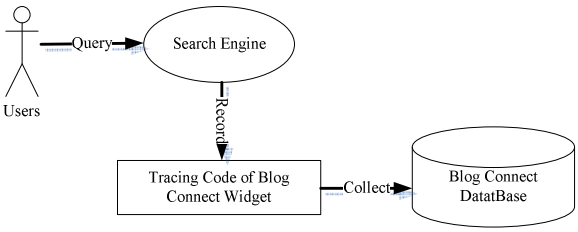

Fig. 1. Tracing Code of BC Widget operator outline

If users enter queried keywords on a search engine and then visit blogs with BC widgets, the tracing code embedded in a BC widget will collect information, such as queried keywords, stay time, and click times, incoming URLs, etc., and then save these information in the database.

The steps of collecting queried keywords are described as follows:

Step 1: A user enters keywords on a search engine to query all related articles. For example, in Figure 2, the entered keywords are "I/O列表 lalaalisa."

Fig. 2. Query example

Step 2: The search engine returns a list of search results and the user clicks on links that may fulfill their intentions, as shown in Figure 3.

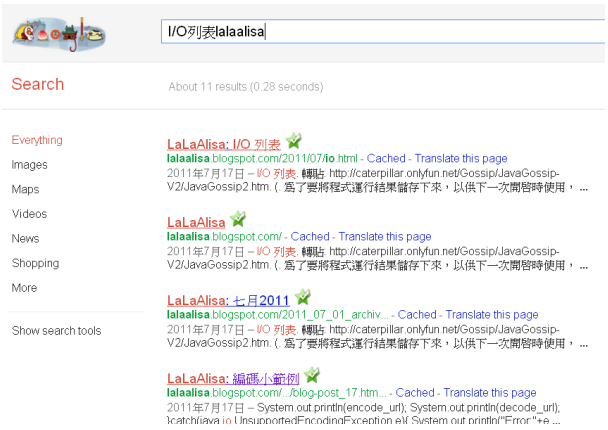

Fig. 3. Search results example

Step 3: After clicking on a link, the search engine redirects the user to a blog which contains a BC widget (the box as shown in the right hand side of Figure 4). The tracing code embedded in the BC widget records the queried keywords, the URL and title of the blog, stay time, etc.

Fig. 4. Blog with BC Widget

In Table 1, the headers "Blog article URL," "Blog article title," "Stay_time," and "Keywords" represent the visiting URL, the title of the article, the total stay time in seconds, and queried keyword; respectively. To be able to process Chinese symbols and English words, text tokenization and word segmentation are required. In this project, mmseg4j [11] based on the MMSeg algorithm [16] was used to tokenize and segment Chinese symbols and English words.

TABLE 1
PART OF THE BLOG CONNECT DATABASE

Blog Article URL	Blog Article Title	Stay_Time	Keywords
http://plane0747.pixnet.net/blog/post/25811198	DAO – Data Access Object 用後心得 @ 空港 Airport :: 痞客邦 PIXNET ::	13	高格 .dat 轉 access
http://colaking.pixnet.net/blog/post/21884319	電影彩色戰士 主題曲+原聲帶試聽 @ Hi! Cloaking :: 痞客邦 PIXNET ::	30	餵耳朵 餵眼睛 印尼

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

In general, to determine which keywords can be used to represent an article, the whole content of the article has to be read, tokenized, and segmented. This process is called full-text keyword retrieval process. Then, based on the frequency of each keyword, keywords that represent the main topic (or theme) of the article are selected. To verify whether or not the queried keywords can also be used to represent the topic of a blog article, the matched ratio (called *m-ratio*) of queried keywords versus the keywords extracted from full-text keyword retrieval process has to be calculated. In this project, the full-text keyword retrieval process consists of the followings which are illustrated in Figure 5:

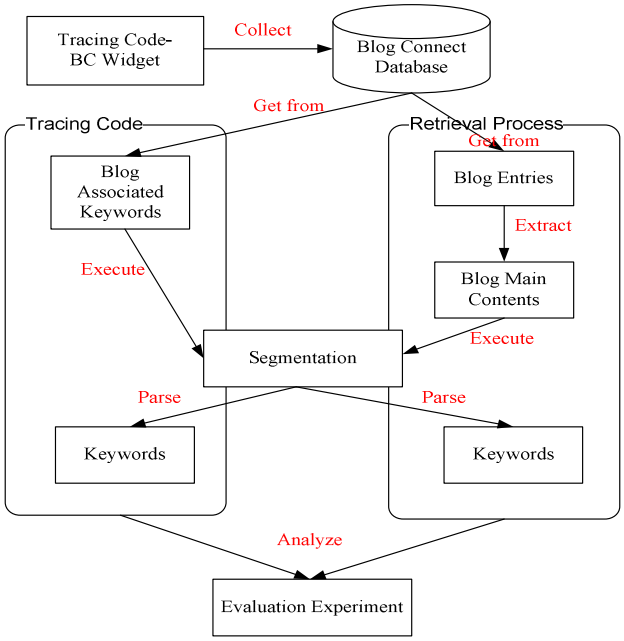

Fig. 5. Overview of the Verified Process

1. The URL of each blog article stored in Blog Connect database is retrieved.
2. Based on the retrieved URLs, the content of blog articles are downloaded.
3. JSOUP parser [14] was utilized to remove HTML tags in the content of each blog. An example blog article is shown in Figure 6.
4. mmseg4j [11] was used to tokenize and segment Chinese symbols and English words. Figure 7 shows the tokenization and segmentation result of the example blog article as shown in Figure 6.

As shown in the left hand side of Figure 5, queried keywords for an article can be retrieved from Blog Connect database. After tokenization and segmentation, a set of queried keywords for each blog article is collected.

Finally, the *m-ratio* of queried keywords versus the keywords extracted from full-text keyword retrieval process has to be calculated. The *m-ratio* is defined as follow:

$$m-ratio = \frac{N_i}{T_i} \qquad (1)$$

where i represents the *i*-th article, T_i is the total number of queried keywords of the *i*-th article and N_i is the number of matched queried keywords against the extracted keywords from full-text keywords retrieval process. Each article has a *m-ratio* so the average *m-ratio* was computed as Equation (2), where *TN* is the total number of articles in Blog Connect

$$Average\ (m-ratio) = \frac{\sum_{i=1}^{TN} m-ratio}{TN} \qquad (2)$$

For example, the followings describe the steps required to calculate the m-ratio for an example article A whose URL is "http://nondescript-hua.blogspot.com/2010/08/ubuntuubuntuserver-1.html":

Table 2 summarized keyword related information of article A. The column "Full-text keyword retrieval" lists the extracted keywords of article A. The columns "Queried keywords" and "The frequency of queried keywords" list the queried keywords and their corresponding frequency; respectively. The columns "Queried keywords after segmentation" and "The frequency of queried keywords after segmentation" list the queried keywords after segmentation and their corresponding frequency; respectively. As shown in the table, the frequencies as shown in column 3 and 5 may be effected by tokenization and segmentation. Take the queried keyword "ubuntu10.4" as an example. Before segmentation, its frequency is 2. However, after segmentation, "ubuntu10.4" became two keywords: "ubuntu10" and "4". Because there is already a queried keyword "ubuntu10" whose frequency is 1, the frequency for "ubuntu10" is now 3.

TABLE 2
THE KEYWORD RELATED INFORMATION OF ARTICLE A

Full-text Keyword Retrieval	Queried Keywords	Frequency of Queried Keywords	Queried Keywords after Segmentation	Frequency of Queried Keywords after Segmentation
tomcat6	tomcat6	3	Tomcat6	3
ubuntu10	ubuntu10.4	2	ubuntu10	3
			4	2
4	Ubuntu	2	Ubuntu	2
Ubuntu	環境變數	1	環境變數	1
設定	6u21	1	6u21	1
apache	i586	1	i586	1
linux	ubuntu10	1		
jdk	apache	1	apache	1
啟動	變數	1	變數	1
變數	linux	1	linux	1
tomcat	tomcat	1	tomcat	1
…	…	…	…	…

The total number of queried keywords after segmentation of article A is 19. Comparing these 19 keywords to the keywords extracted from full-text keyword retrieval (listed in column 1), there are 18 matching keywords. Thus, m-ratio of article A is calculated as follow:

$$m-ratio = \frac{18}{19} = 0.9474$$

IV. EXPERIMENTS

The total number of BC articles collected so far was 398, and their *m-ratio* distribution is shown in Figure 8. As shown in Figure 8, there are 37 articles whose *m-ratio* is 0. After investigation, it is found that queried keywords are not

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

contained in the content of those articles, but in the sidebar of the corresponding blog pages which resulted in their *m-ratio* being equal to 0; after removing the abnormal data, the average *m-ratio* o was 0.824.

When the frequency of a queried keyword is larger than other queried keywords, it is believed the queried keyword is more important than the others. Therefore, the threshold *t* was defined to filter out unimportant keywords. After filtering out those *m-ratio*s whose *t*=1, the number of articles is 225 and the *m-ratio* increased to 0.892, and the *m-ratio* distribution is shown in Figure 9.

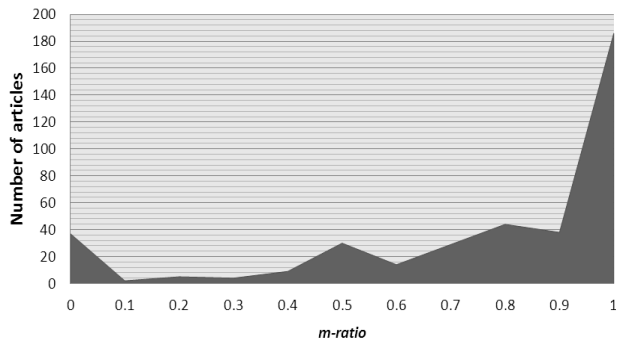

Fig. 8. *Match Ratio* Distribution of 398 BC Articles

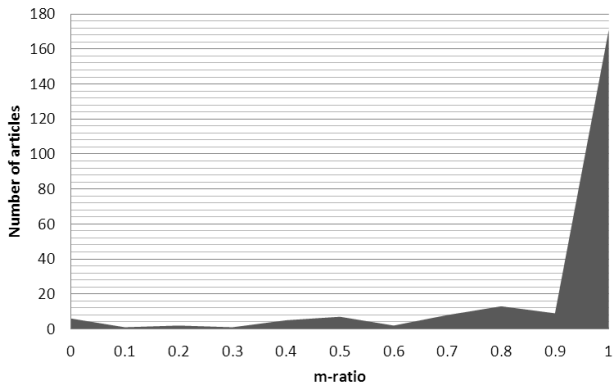

Fig. 9. *m-ratio distribution if t >1*

Without considering the time required to download articles, the time needed to generate keywords by using full-text keywords retrieval process and the proposed scheme is listed in Table 3.

TABLE 3
TIME COST OF TWO METHOD COMPARED

		Proposed Scheme		Full-text Keywords Retrieval	
		Input Text Length	Time Cost (Sec)	Input Text Length	Time Cost (Sec)
Part I	Total	-	-	-	152.589
	Average	-	-	-	0.383
Part II	Total	12671	0.218	421200	0.703
	Average	32	5.477E-4	1058	0.002

The keyword generation was divided into two parts, removing the html tags and segmenting the terms (part I and part II, respectively). In part I, no processing was needed by the proposed scheme. The total time and average time for processing part I using full-text keywords retrieval process was 152.589 seconds and 0.383 seconds, respectively. The total processing time needed in part II for the proposed scheme and full-text keywords retrieval process was 0.218 and 0.703 seconds. On average, the processing time needed in part II for the proposed scheme and full-text keywords retrieval process was 5.477E-4 and 0.002 seconds; respectively.

V. CONCLUSIONS AND FUTURE WORK

With the tremendous amount of online blog content, users need an effective way to read and obtain information. One approach to provide better information sharing is to create relationships between related blogs. A general approach to determine whether or not two blogs are related to each other is to extract keywords through full-text keyword retrieval process. However, this process is complex and time-consuming. This paper showed that queried keywords can be used to represent the main topic of a blog article, instead of using keywords extracted from full-text keyword retrieval process. The *m-ratio* is up to 0.892, and time was saved compared with the general approach. Therefore, the positive data confirmed that the proposed scheme is a feasible solution. In future work, we will further consider the relationships between keywords selection and stay time, and incorporate WordNet [17] into the scheme to enhance the precision of keyword selection.

REFERENCES

[1] N. Ali-Hasan and E. Adamic, “Expressing Social Relationships on the Blog through links and comments,” Available: www.ladamic.com/work/papers/oc/onlinecommunities.pdf (accessed 9 June 2008).
[2] Blog Connect, Available: http://bridge.nchu.edu.tw/BC/.
[3] Blogster, Available: http://www.blogster.com/.
[4] J. L. Elsas, J. Arguello, J. Callan and J. G. Carbonell, “Retrieval and Feedback Models for Blog Feed Search,” in *Proc. 31st annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, Singapore, July 2008, pp. 347-354.
[5] J. Gao and W. Lai, “Formal Concept Analysis Based Clustering for Blog Network Visualization,” in *Proc. International Conference on Advanced Data Mining and Applications, Berlin: Heidelberg*, 2010, pp. 394-404.
[6] G. Hope, T. Wang, and S. Barkataki, “Convergence of Web 2.0 and Semantic Web: A Semantic Tagging and Searching System for Creating and Searching Blogs,” in *Proc. IEEE International Conference on Semantic Computing (ICSC)*, Irvine: California, 2007, pp. 201-208.
[7] B. J. Jansen, D. L. Booth, and A. Spink, “Determining the User Intent of Web Search Engine Queries,” in *Proc. International Conference on World Wide Web, Alberta: Canada*, 2007, pp.1149-1150.
[8] L. Lu and F. Zhu, “Blogger clustering by utilizing link information,” in Proc. IEEE International Conference on Intelligent Computing and Intelligent System(ICIS), Xiamen, China, Oct. 2010, pp. 267-270.
[9] N. Johnson, 2008, “Google on User Intent in Search Queries, Search Engine Watch,” [Online] Available: http://searchenginewatch.com/article/2053806/Google-On-User-Intent-in-Search-Queries.
[10] A. Juffinger and E. Lex, “Crosslanguage Blog Mining and Trend Visualisation,” in *Proc. 18th International World Wide Web Conference, Madrid, Spain*, 2009, pp.1149-1150.
[11] mmseg4j, Available: http://code.google.com/p/mmseg4j/.
[12] A. Qamra, B. Tseng, and E. Y. Chang, “Mining Blog Stories Using Community-based and Temporal Clustering,” in *Proc. 15th ACM International Conference Information and Knowledge Management, Arlington: Virginia, USA*, 2006, pp. 58-67.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[13] J. Sobel, (2010, Nov. 3). "State of the Blogosphere 2010 Introduction. Technorat," Available: http://technorati.com/blogging/article/state-of-the-blogosphere-2010-introduction/.
[14] R. Stokes, *Ultimate Guide to Pay-Per-Click Advertising*, Irvine, CA: Entrepreneur Press, 2010.
[15] Touchgraph, Available: http://www.touchgraph.com/navigator.
[16] C.-H. Tsai, (2004, Feb. 5). "A World Identification System for Mandarin Chinese Text Based on Two Variants of the Maximum Matching Algorithm", Chih-Hao Tsai's Technology Page [Online]. Available: http://www.geocities.com/hao510/mmseg/.
[17] WordNet, Available: http://wordnet.princeton.edu/.
[18] Y. Zhang, K. GAO, B. Zhang, J. Guo, F. Gao, and P. Guo, "Clustering Blog Posts Using Tags and Relations in the Blogosphere" in *Proc.* 1st International Conference on Information Science and Engineering(ICISE), Nanjing, China, Dec. 2009, pp. 817-820.

ISBN: 978-988-19251-1-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)