# Research and Implementation of Index Real-time Updating Based on Lucene

Nana Zhang [a], Yisong Wang and Kun Zhu
College of Computer Science and Technology, Guizhou University, China
[a]nana_zhang92@163.com

**Keywords:** Lucene, Real-time index, Index library, Word cloud.

**Abstract.** Before establishing the index, it is necessary to delete all the index files that created previously in the index library and then re-establish the index, which is time-consuming. For this problem, this paper proposed an index real-time updating method, and implemented that showed the content of the document in the form of word cloud, and then decided whether to put the document into the index library. Experiments showed that the time of establishing the index had been greatly improved.

## 1. Introduction

Since the 21st century, the Internet has entered the stage of rapid development, the information and resources on the Internet has become increasingly rich, the amount of information resources grow rapidly in the form of exponential level. Therefore, if want to retrieve the useful information quickly and accurately from a large amount of resources, we need to choose the right search engine tool [1, 2].

Lucene is an open source full-text search engine toolkit that includes both establishing indexes and retrieving[3].Before searching, because people do not know which documents have been indexed, which documents are not indexed, so they will delete all index files in the index library, then index all documents, which is time-consuming. If not delete the previous index files, and index directly, in that way, it will retrieve repeated documents, because indexing the document about two or more times. In view of this problem, this article implemented the index real-time updating. And if you want to decide whether to index documents based on the content, this paper implemented that showed the content of the document by means of the word cloud.

## 2. Lucene Overview

### 2.1 Lucene introduction.

Lucene uses an inverted index to establish indexes [4].Lucene uses a platform-independent index file format, and Lucene customizes the unique index file format and stores all the index files in the same format. Therefore, users can share the resources of the index files on different platforms [5].

The key advantages of the Lucene search engine toolkit are that the source code is open [6] and convenient for secondary development [7], moreover, the index structure is unique and the engine architecture is well designed.

The main three aspects about Lucene's application scope:

①To design and implement search engine based on Lucene;

②Add technology based on Lucene in all kinds of software systems, the combination of both[8];

③Add the Lucene search engine to the web.

### 2.2 Lucene system structure framework.

Lucene has better object-oriented capability, and it is easy to achieve Lucene's secondary development. Lucene system architecture framework is shown in Fig. 1[9]:

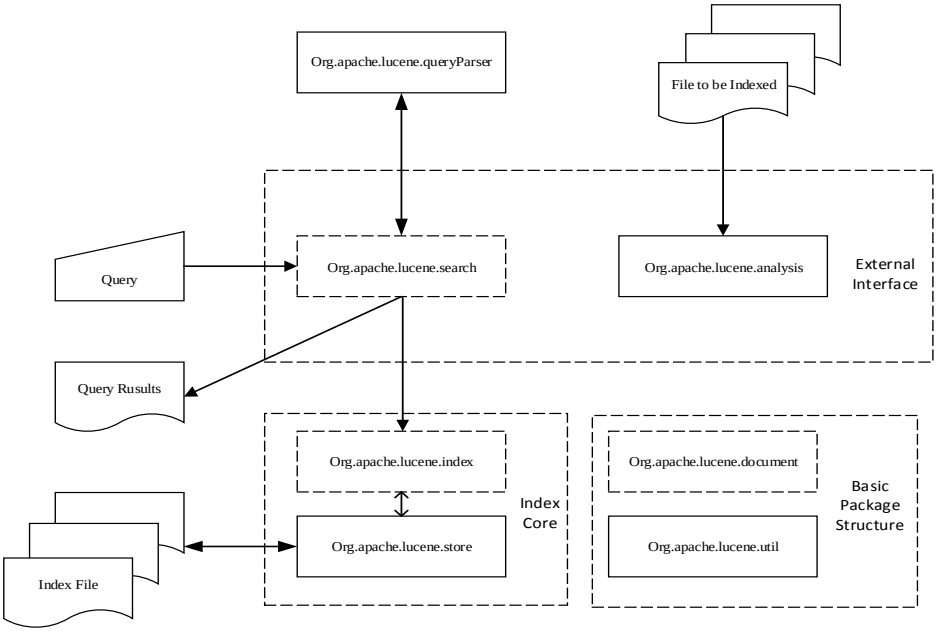


Fig. 1 Lucene system architecture framework

As shown in Fig.1, Lucene search engine toolkit mainly includes three parts [10]: (1) input/output interface; (2) index core; (3) basic package structure framework.

## 3. Index Real-time Updating

### 3.1 Experiment Environment.

MyEclipse development environment: JDK1.7.0_67 version, lucene_30 version, download xmlbeans-2.3.0.jar, lucene-core-2.0.0.jar, PDFBox-0.7.3.jar, jsoup-1.7.2 and other required jar packages, a large number of documents in local disk, and the class in the MyEclipse calls R software which is 3.2.2 version.

### 3.2 Contrast Environment.

3.2.1 before Improvement.

Delete all the previous index files, and then re-index all the types of documents. As shown in Fig. 2.

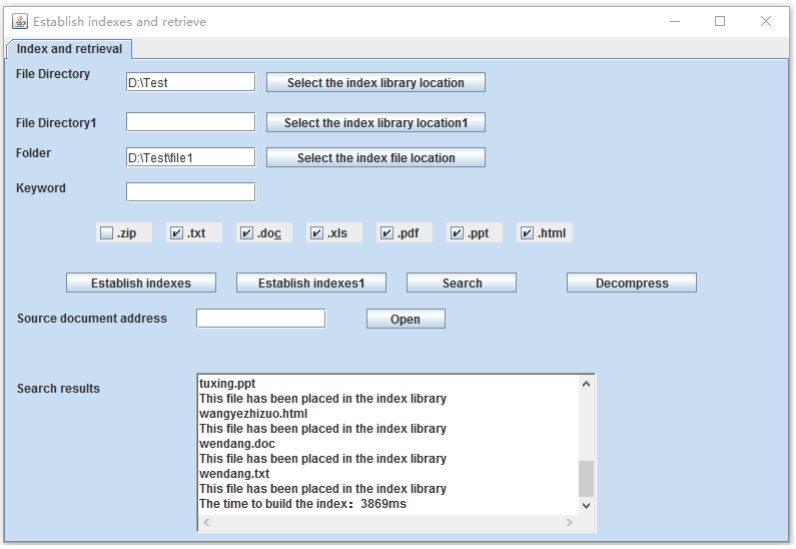


Fig. 2 The execution result before improvement

If the index files are not all deleted in the index library before re-establishing indexes, then the same document will be retrieved repeatedly. As shown in Fig. 3.

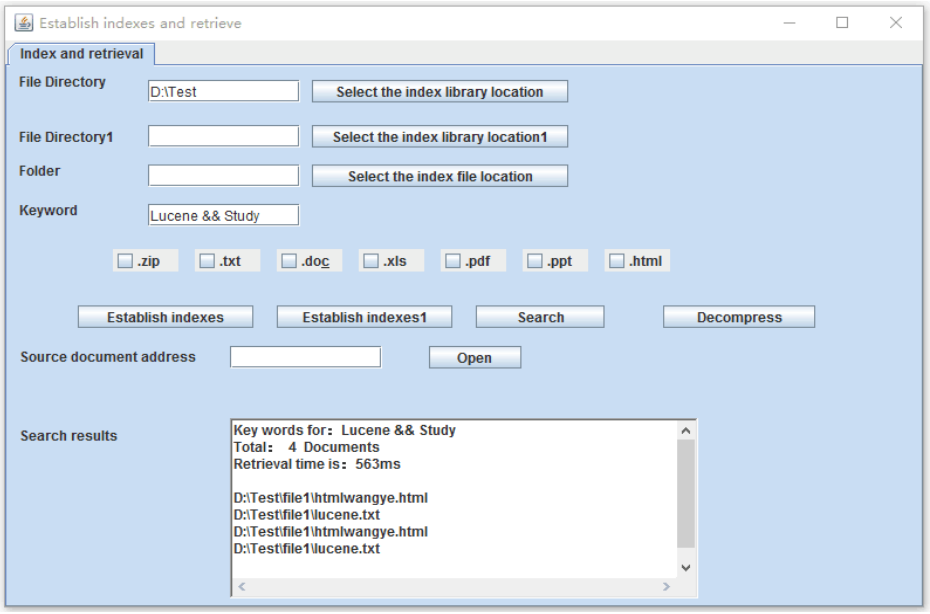


Fig. 3 The result of retrieving documents repeatedly

3.2.2 After improvement.

When establish indexes, click the "Select the index library location" button firstly, display the last indexed time and now indexing time; then click the "Select the index file location" button, and display the last modification time for all documents in this folder, and which types of documents have been indexed in this index library. As shown in Fig. 4.

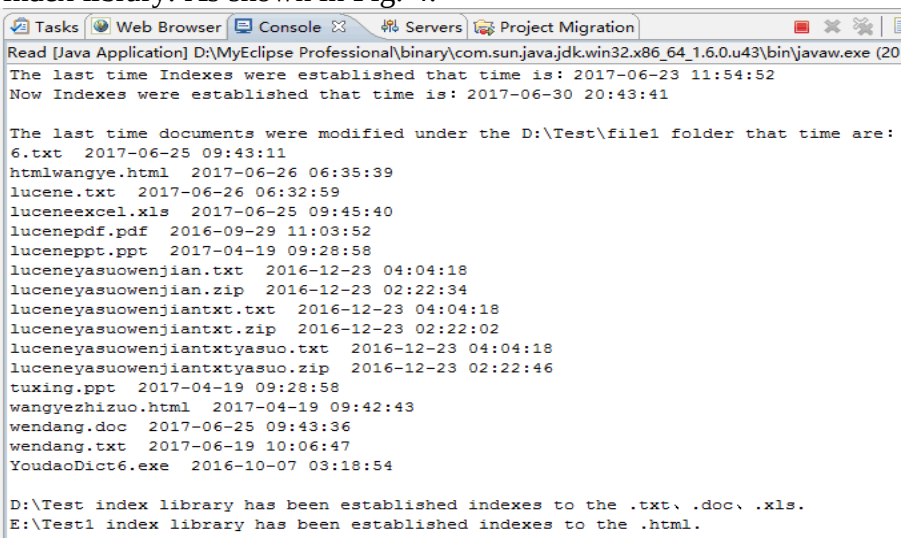


Fig. 4 Index real-time updating

By clicking the button twice, we can get what documents are newly added after the last index and what types of documents have been indexed. Therefore,we create a new folder at first, and the new document will be added to this folder. When implemented we click the "Select the index file location" button, that is, we select the location of the new folder, and then we index the add documents under this folder, greatly improving the speed of indexing.

3.2.3 Experiment results.

Add the new three articles about 6.txt, luceneexcel. xls and wendang.doc or four articles about 6.txt,luceneexcel.xls,wendang.doc and lucentppt. ppt or five articles about 6.txt,luceneexcel.xls, wendang. doc, lucentppt.ppt and wangyezhizuo.html or six articles about 6.txt, luceneexcel. xls, wendang. doc,lucentppt.ppt,wangyezhizuo.html and lucenestudy.pdf in the D:\Test\file1 folder, the contrast of experimental results is shown in Table 1 and Fig. 5.

Table 1 The time comparison table

| The number of newly added documents(piece) | three | four | five | six |
|---|---|---|---|---|
| Before improvement(ms) | 3869 | 4359 | 4641 | 5313 |
| Real-time(ms) | 2328 | 1781 | 1862 | 2699 |

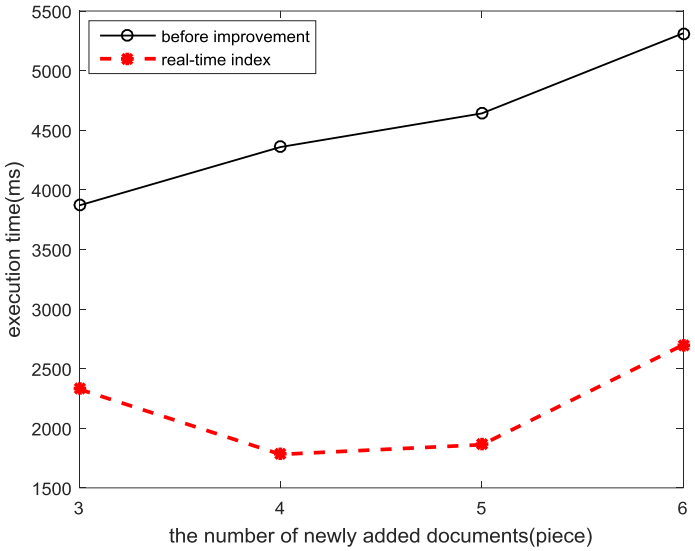


Fig. 5 The time comparison chart

As shown in Table 1 and Fig. 5, as the number of newly added documents increases, both the time of establishing index are increasing, but after improvement the time of establishing index is significantly more lower.

## 4. Word Cloud

### 4.1 Word Cloud Introduction.

"Word cloud" is the frequency is higher in the network text of the "keywords" to be the highlight on the vision, forming "keywords cloud" or "keywords rendering", so as to filter out a lot of text information, make people who sweep text with only a glance that can understand the main idea of the document.

"Word cloud" can give visually highlighted show for the higher frequency "keyword" in the network text, which forms "keyword cloud" or "keyword rendering", and filter out a lot of text information, so people can understand the subject of the document quickly.


Fig. 6 The word cloud form of the document

### 4.2 The word cloud form of the document.

We can use the word cloud to understand the main content of documents, and then decide whether to index the document.

If we want to understand the main content of the D:\Test\file1\wendang.doc, the word cloud form of the document is shown in Fig. 6:

Through the word form of the document, we can understand that the main content of the document is about index, search, data, type, so can decide whether to add the document to the index library as required.

## 5. Summary

This paper shows the content of the document through the form of word cloud, and it is easy to understand the main content of the document, then decide whether to index the document. Combined with Lucene full-text search framework, and avoid indexing all the documents under the folder through the implementation of real-time index, experiments verify that this approach improves the speed of establishing indexes. The next step, we will consider that if a document have been indexed and then revise to this document, how to implement the real-time index of such a document.

## Acknowledgements

This work was supported by the General Program of National Natural Science Foundation of China (NSFC) under Grant No.60963009.

## References

[1]. Michael McCandless, Erik Hatcher.Lucene in action (second edition).American: Manning Publications, 2011.

[2]. B. Li, J. Zhang, M. Chen, J. Zhang, K. Wang and D. Meng, "DIFTSAS: A DIstributed Full Text Search and Analysis System for Big Data," 2013 IEEE 16th International Conference on Computational Science and Engineering, Sydney, NSW, 2013, pp. 1303-1309.

[3]. J. Cao, J. Lin, S. Wu, M. Guan, Q. Dai and W. Feng, "Lucene and deep learning based commodity information analysis system," 2016 IEEE International Conference on Consumer Electronics-China (ICCE-China), Guangzhou, 2016, pp. 1-4.

[4]. G. He, Y. Xiao, D. Yu and X. Wu, "Enterprise Network Status Analysis Using Hadoop and Lucene," 2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, 2015, pp. 527-530.

[5]. X. Chen and L. Xu, "An Educational Resource Retrieval Mechanism Based on Lucene and Topic Index," 2016 13th Web Information Systems and Applications Conference (WISA), Wuhan, 2016, pp. 125-128.

[6]. Y. Zhou, X. Wu and R. Wang, "A semantic similarity retrieval model based on Lucene," 2014 IEEE 5th International Conference on Software Engineering and Service Science, Beijing, 2014, pp. 854-858.

[7]. H. Chen et al., "BloomCast: Efficient and Effective Full-Text Retrieval in Unstructured P2P Networks," in IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 2, pp. 232-241, Feb. 2012.

[8]. J. Chen, W. Wu and C. Wang, "A mobile phone information search engine based on Heritrix and Lucene," 2012 7th International Conference on Computer Science & Education (ICCSE), Melbourne, VIC, 2012, pp. 1602-1604.

[9]. X. Shi and Z. Wang, "An Optimized Full-Text Retrieval System Based on Lucene in Oracle Database," 2014 Enterprise Systems Conference, Shanghai, 2014, pp. 61-65.

[10]. Liang Cao, Weiming Wu and Yonghao Gu, "The research of performance of Lucene's Chinese tokenizer," 2011 2nd International Conference on Artificial Intelligence, ManagementScience and Electronic Commerce (AIMSEC), Deng Leng, 2011, pp. 7398-7401.