Available online at www.sciencedirect.com

ScienceDirect

Procedia Social and Behavioral Sciences 18 (2011) 282–286

Procedia Social and Behavioral Sciences

Kongres Pengajaran dan Pembelajaran UKM, 2010

Development of Search Engines using Lucene: An Experience

Masnizah Mohd*

Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Malaysia

Abstract

Lucene is a Java library which is able to perform the indexing and searching process. It allows the development of a text-based information retrieval systems or applications such as search engines. This paper intends to discuss the issues, and share the experience of using Lucene during course project development. Lucene is used by 28 second year students who are in the Information Science programs. They have to implement Lucene library in the Development of Search Engines (TP2433) course project. Results from the analysis have contributed in providing guidelines for future handling of the final TP2433 project.

Selection and/or peer-review under responsibility of Kongres Pengajaran & Pembelajaran UKM, 2010

Keywords: Lucene; Information Retrieval; Search Engines;

1. Introduction

Information retrieval (IR) is a field concerned with the structure, analysis, organization, storage, searching, and retrieval of information (Salton 1968). It emphasizes on the process of matching user queries to the index in finding relevant documents. In fact, the main issue in this area is to ensure a good match with high similarity score by comparing between the queries and the document index. Search engines such as Google are the practical applications of IR techniques on large-scale text collections.

Search engines should include the concept, models, techniques and the processes of IR. Two major components of search engines are the indexing and query processes (Croft *et al*. 2010). The indexing process aims to create data structures or the indexes that allows the searching. Meanwhile the querying process will use the structures and user queries to generate a ranked list of documents. Figure 1 depicted the indexing process in search engines. It involves three components; text acquisition, text transformation and index creation as described in Table 1.

* Corresponding author. Tel.: +0-603-8921-6671; fax: +0-603-8926-7950
E-mail address: mas@ftsm.ukm.my

1877–0428 © 2011 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.sbspro.2011.05.040

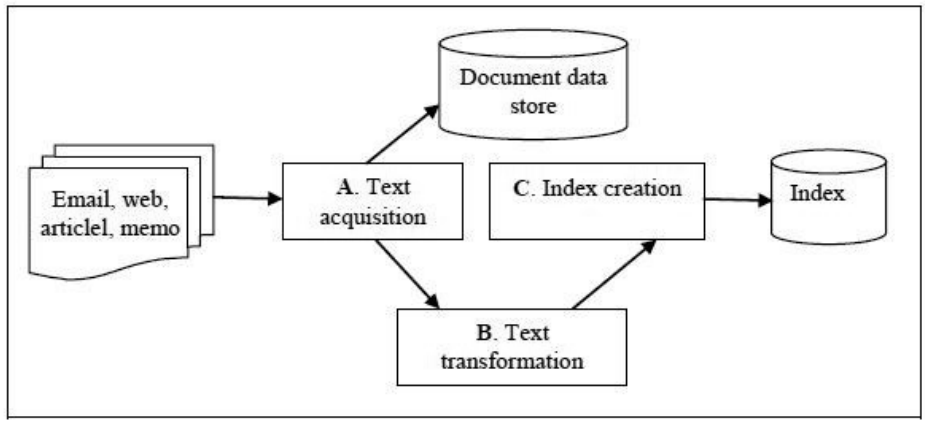

Figure 1 Indexing process in search engines

Table 1. Three components of the indexing process in search engines

	Process	Description
A.	Text acquisition	Identifies and stores documents for indexing. Documents are in various formats such as email, websites, memos, letters and articles.
B.	Text transformation	Transforms documents into index terms or features which involves lexical analysis (*parsing-tokenizing-stopword removal-stemming*).
C.	Index creation	Takes index terms and creates data structures (indexes) to support fast searching

Figure 2. shows the query process in search engines. It involves three components; user interaction, ranking and evaluation as indicated in Table 2.

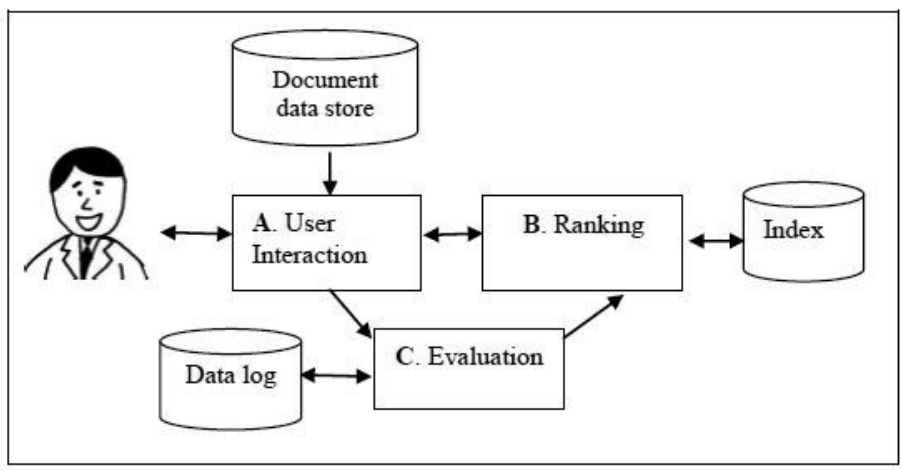

Figure 2 Query process in search engines

Table 2. Three components of the query process in search engines

Process	Description
A. User interaction	Supports creation and refinement of query, display of results.
B. Ranking	Uses query and indexes to generate ranked list of documents.
C. Evaluation	Monitors and measures effectiveness and efficiency (primarily offline)

2. Lucene

Lucene is an open source Java library, which supports the process and techniques of information retrieval. Applications such as Amazon are among the commercial application that uses Lucene for indexing and allowing effective searching. Lucene is able to index text from a various formats such as PDF, HTML and Microsoft Word, and also in various languages. The key classes used to build search engines are (Paul 2004):

a. ***Document*** - The *Document* class represents a document in Lucene. We index *Document* objects and get *Document* objects back when we do a search.
b. ***Field*** - The *Field* class represents a section of a *Document*. The *Field* object will contain a name for the section and the actual data.
c. ***Analyzer*** - The *Analyzer* class is an abstract class that is used to provide an interface that will take a *Document* and turn it into tokens that can be indexed. There are several useful implementations of this class but the most commonly used is the *StandardAnalyzer* class.
d. ***IndexWriter*** - The *IndexWriter* class is used to create and maintain indexes.
e. ***IndexSearcher*** - The *IndexSearcher* class is used to search through an index.
f. ***QueryParser*** - The *QueryParser* class is used to build a parser that can search through an index
g. ***Query*** - The *Query* class is an abstract class that contains the search criteria created by the *QueryParser*.
h. ***Hits*** - The *Hits* class contains the *Document* objects that are returned by running the *Query* object against the index.

Lucene supports an extensive search criteria such as Boolean (AND, OR and NOT), fuzzy searches, proximity searches, wildcard searches, and range searches. Some examples of searches are:

i. Find all Masnizah articles related to information retrieval and search engines:
author:Masnizah information retrieval AND search engines
ii. Find all articles that contain the phrase clustering and exclude Single Pass
clustering NOT Single Pass
iii. Find all articles written by Masnizah in February this year
author:Masnizah date:[02/01/2011 TO 02/28/2011]

3. Methodology

TP2433 course is compulsory for the second year Information Science students at the Faculty of Information Sciences and Technology (FTSM). It was offered in semester 1. A total of 28 students worked in group and they were the first batch to use Lucene. They were asked to use Lucene in their project with the evaluation percentage of 30%.

The students have to develop an IR system or search engines that apply the IR techniques learnt during lectures. Thus, students' understanding of the IR concept can be enhanced through the use of Lucene during the project implementation. For example, students can increase their understanding of the indexing process to construct the index using Lucene. Assessment of the project is divided into two categories:

i. Presentation (10%)
   a. The ability to explain the program.
   b. The ability to answer question from the lecturer and colleagues

ii. Development of search engines (20%)
   a. The usage of classes in Lucene
      This is to evaluate the students' ability to use and be able to customize classes in Lucene in their project
   b. Index creation
      The quality of index created after the lexical analysis process
   c. The ability of searching process
      The various searching criteria offered such as allowing the Boolean and wildcard is an advantage.
   d. The accurate results
      The ability of search engines to return an accurate and relevan results based on user query.

Students were given a set of questionnaires after the TP2433 course. The questionnaire aims to validate students' achievement on the learning outcomes based on their understanding. The scale used in the questionnaire was:

1- Unachieved
2- Less achieved
3- Average
4- Achieved
5- Well achieved

4. Results

We analysed the students' achievement on the learning outcomes based on their understanding. The result was shown as in Table 3.

Table 3. Percentage of students' achievement on the learning outcomes based on their understanding

Learning Outcome (LO)			Scale				
			1	2	3	4	5
LO1	1.	Able to define the concept of information retrieval (IR) and search engines (SE).	0.0	0.0	0.0	32.1	67.9
LO2	2.	Able to identify the components, techniques and models of IR.	0.0	0.0	3.6	25.0	71.4
LO3	3.	Able to explain the process of an IR system.	0.0	0.0	17.9	25.0	57.1
LO4	4.	Able to identify, analysis and evalute the effectiveness of search engines.	0.0	0.0	0.0	7.1	92.9
LO5	5.	Able to develop search engines or an IR system using the principles and techniques learned.	3.6	10.7	28.6	46.4	10.7

Most of the students agreed that they have achieved the first learning outcome (LO1) where they were able to define the concept of IR and SE. 67.9% of students agreed that LO1 was well achieved (scale 5) and 32.1% of students agreed that it was achieved (scale 4).

Meanwhile for the second learning outcome (LO2) on students' ability to identify the components, techniques and models of IR , there were 71.4% of students agreed that the LO2 was well achieved (scale 5) and 25% of students agreed it was achieved (scale 4).

Findings also indicated that most of the students agreed that they have achieved the third learning outcome (LO3) on the ability to explain the process of an IR system. 57.1% of students agreed that LO3 was well achieved (scale 5) and 25% of students agreed that it was achieved (scale 4).

The fourth learning outcome (LO4) received the highest percentage with 92.9% of students agreeing that it was well achieved. This indicates that the students were able to measure an effective and efficient search engines.

The focus of this paper is on the fifth learning outcome (LO5) which is on the ability to develop search engines. Results showed that the proportion between the number of students who agreed (scale 4 and 5) with the number of students who are less agreed (scale 1 and 2) on the ability to develop a search engine using the principles and techniques learned, is 4 to 1 ratio. It was observed that students have misperception of Lucene where they thought it was an IR system. Gradually students solved these problems through an explanation and demonstration on the implementation of Lucene in the final project. Based on the project presentation, students have used 70% of the primary classes in Lucene but the ability to develop a search engine can still be improved. The quality of the index created also can be improved because there is no application of the stemming process. Almost 60% of students used the exact match and applied the use of Boolean operators in the search process. Finally, the search engines have returned relevant documents based on user query where the overall accuracy was still at a moderate level.

These are the guidelines for the future conduct of TP2433 course:

i. TP2433 course offering
   This course should be offered in the second semester after the students already have a good Java programming skill. This is because the usage of Lucene relies most on the Java skills. Lucene only provide the classes to perform the IR techniques.
ii. Course basic requirement
   TP2433 should have the Java programming course as the basic requirement. Student who did not have Java programming background will face difficulties to understand Lucene framework. It was observed that they have a low programming skill, thus it has affected their contribution to the project.

5. Conclusion

Lucene has been introduced for the first time in TP2433 course. Lucene is the *state of the art* in document preprocessing and in index creation. Thus, it was relevant to enforce the students to use Lucene. It will increase their understanding of the IR theory and approaches in developing a search engine. Students should be able to see the practicality of IR techniques by using Lucene. This paper has identified the requirement for the course project development and therefore has come out with a guideline for future conduct of TP2433 course.

6. Acknowledgement

The author would like to thank the Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia for the opportunity given in conducting this research.

References

Croft, W.B., Metzler, D. & Strohman, T. (2010). *Search Engines: Information Retrieval in Practice*. London: Pearson.
Paul, T. (2004). The Lucene Search Engine. http://www.javaranch.com/journal/2004/04/Lucene.html [2 November 2010].
Salton, G. (1968). *Automatic information organization and retrieval*. New York: McGraw-Hill.