CS410 Technology review for Apache Lucene

by
Gayathri Coimbatore Ramachandran
gc24@illinois.edu

Table of contents

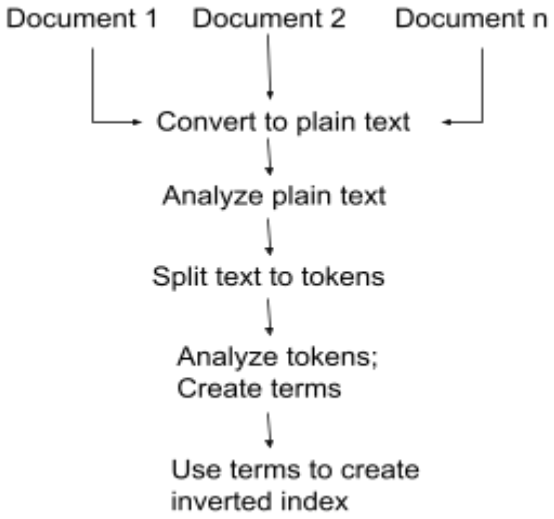
1. Introduction

Apache Lucene is an open source full-featured text search engine software library written in Java. It is a very powerful and an extremely fast library that is well suited for any application that requires full-text search. Lucene's accurate and efficient search algorithms offer a seamless search experience for the users. The indexing and the search technology allows the implementation of full-text search capabilities and efficient indexing processes on applications in a cross platform environment.

2. Searching using Lucene

Lucene is able to achieve fast search responses since it uses inverted indices instead of creating a classic index. The classic index technique usually collects the full list of words the document contains for every document. But Lucene uses inverted indices which means for every word in all the documents, Lucene stores what document and position the particular search word/term can be found at. This makes the search easier and quicker, since Lucene uses index to search instead of searching the text on all the documents.

3. Lucene Workflow



As seen in the diagram above:

1. The documents are fed to Lucene
2. For every document, Lucene first reads the text sources and converts it into plain text. Runs analyzers to analyze the plain text and split the text into tokens. An analyzer internally uses a tokenizer. It does more than simple text splitting. It performs different kinds of text analysis like
    - Stemming: Replace words with their stems
    - Stop Words Filtering: Removing certain noise words that are of no particular interest when performing a search.
    - Text Normalization: Remove accents and other character markings
    - Synonym Expansion: Add synonyms at the same token position as the current word.
3. Analyze tokens to create terms
4. Finally use the terms generated in the previous step to create the inverted index

Now the index is ready to be searched. Users can write queries in different formats and Lucene returns all the documents that match the query.

4. Features

Lucene offers the following powerful features through a simple API:

4.1 Incremental Indexing

Lucene offers a feature called incremental indexing that allows users to add, change or delete documents without the need to re-index the documents. Since Lucene uses timestamps to identify the document changes, only those appropriate changes will need to be re-indexed. This feature is very useful when dealing with large databases since it is as fast as batch indexing.

4.2 Scalable High Performance Indexing

Apache Lucene is highly scalable. It lets users process more documents and index them in less time. The search engine library supports indexing of over 50 GB of data

in one hour. Lucene has small RAM requirements, only about 1 MB of heap. The index size is roughly 20-30% the size of text indexed.

4.3 Efficient Search algorithms

Lucene provides a highly configurable hybrid form of search that combines exact boolean searches with relevance ranking oriented vector space search methods. In addition the search engine library allows us to use different ranking models such as Vector Space Model and Okapi BM25.

4.4 Powerful, Accurate and Efficient search

Lucene makes it easier for users to find any information they need. It supports many powerful query types like phrase queries, wildcard queries, range queries, proximity queries and more. Accurate searching can be implemented by enabling field search functionality. The most common types of fields are author, title, subject, abstract etc. Searching by fields allows Lucene to look for information from a specific area hence making search faster and more efficient. It also allows sorting by any field. Lucene supports multiple index searching with merged results. With this capability, pieces of information are searched on multiple indices all at once. The data it gathers from each index can be combined together generating merged search results. Another interesting feature is that it allows simultaneous updating and searching. All of these features make Lucene a very powerful, accurate and efficient search library.

4.5 Cross-Platform Solution

Lucene is written in Java and is available as Open Source software. Hence could be used both in commercial and Open Source Program. Most importantly. It is a cross-platform solution. Therefore, it is popular in both academic and commercial settings due to its performance, reconfigurability and generous licensing.

5. Conclusion

Through the use of powerful, accurate and efficient search algorithms and scalable high performance indexing, Lucene makes searching easy. The users are able to find any information they need since Lucene delivers results exactly what the users are looking for. This makes Lucene's search engine library truly a world class technology that can handle sophisticated queries and generate results accurately.

6. References

https://lucene.apache.org/
https://dzone.com/articles/apache-lucene-a-high-performance-and-full-featured