Thai Text Search with Lucene

Jan 19 2009

หลายวันที่ผ่านมามีเพื่อนนักศึกษาปี 4 ต่างมหาวิทยาลัยเมล์มาถามเรื่องการทำ search engine และสนใจโปรแกรม Sansarn Offline ก็เลยถือโอกาสนี้แบ่งปันความรู้เรื่องการทำ search อย่างง่ายๆที่นี่ละกัน

Sansarn Offline เขียนโดยใช้ภาษา Java และตัว engine ที่ทำ search นั้นก็ใช้ Lucene ซึ่งเป็น library สำหรับใช้ในการค้นคืนเอกสารโดยเฉพาะ ต่อไปนี้เป็น code ตัวอย่างการใช้งาน Lucene กับข้อความภาษาไทย สิ่งที่ต้องใช้มี 2 อย่างคือ

1. Lucene library สามารถหาได้จาก lucene.apache.org เมื่อโหลดมาแล้วจะมีไฟล์ชื่อประมาณ lucene-core-2.x.x.jar ให้เพิ่มไฟล์นี้ลงไปใน Java classpath ด้วย
2. ThaiAnalyzer เป็นตัวตัดคำภาษาไทยเพื่อให้ Lucene ตัดคำไทยได้ จะได้ค้นคืนโดยใช้คำสำคัญ (keyword) ได้ สามารถหาได้จาก http://sansarn.com/look/download.html เมื่อได้มาแล้วให้เพิ่ม ThaiAnalyzer.jar ลงใน classpath

เมื่อได้ 2 อย่างที่กล่าวมาแล้วก็เริ่ม demo กันเลย

```
import java.io.IOException;
import org.apache.lucene.analysis.Analyzer;
import org.apache.lucene.analysis.standard.StandardAnalyzer;
import org.apache.lucene.analysis.th.ThaiAnalyzer;
import org.apache.lucene.document.Document;
import org.apache.lucene.document.Field;
import org.apache.lucene.index.CorruptIndexException;
import org.apache.lucene.index.IndexWriter;
import org.apache.lucene.queryParser.QueryParser;
import org.apache.lucene.search.Hits;
import org.apache.lucene.search.IndexSearcher;
import org.apache.lucene.search.Query;
import org.apache.lucene.store.Directory;
import org.apache.lucene.store.FSDirectory;

/**
* Demonstrate how to use basic features of Lucene
* with THai text.
*
* You can freely modify this file to suit your needs.
* There is no any license nor any guarantee associated
* with this file.
*
* @author Wittawat Jitkrittum
* January 19, 2009 10:30 am
*/
public class LuceneDemo {

/**In this example, we add only 2 documents.*/
public static void addSampleDocuments(IndexWriter writer) throws
CorruptIndexException, IOException{
/**In Lucene, to add contents to the index, you need to
construct a Document object. Most of the time, a document
is one web page. (This is not necessary. It can be one paragraph
* if you want)*/
Document doc1 = new Document();
/**One document consists of a collection of fields. You can think of
Field as a part of a document. In most cases, we use three fields namely,
title field, content field, and url field. You may add more if you want.*/
doc1.add(new Field(
"title", //this is the field name
"Test Lucene.", //the title field
Field.Store.YES, //store the title so that we can retrieve and show later.
Field.Index.TOKENIZED //Tokenize and index the title so that we can search by
keywords.
));

/**Add a content field*/
doc1.add(new Field(
"content",
"ผมสร้างไฟล์นี้ขึ้นมาเพื่อเป็น ตัวอย่างแก่ผู้ที่ต้องการสร้าง search engine อย่างง่ายๆ โดยใช้ Lucene" +
"Lucene เป็น library ที่เข้าใจง่าย ผู้ใช้ไม่จำเป็นต้องรู้รายละเอียดของการทำงานของ search engine
มาก",
Field.Store.COMPRESS, //Compress the content. Most of the time, contents is big.
Field.Index.TOKENIZED
));

/**Add a url field*/
doc1.add(new Field(
"url",
"http://www.thisisasampleurl.com/doc1.html",
Field.Store.YES,
Field.Index.NO //THere is no need to index nor tokenize the url. We only keep it
to show to the users. We don't search the url.
));

//-------------------------------------
/**Make another document*/
Document doc2 = new Document();
```

"I know that I know nothing" -- Socrates

Tags

allnews browser conference database data recovery dynamic dns experiment firefox font fun gadget gimp gwt hosting iphone java javascript jtcc kicss2008 laptop latex library linux lucene lyric myself mysql netbeans news nlp parser plugin python qast quiz semantic web senior project server song tarot tcc thai trip ubuntu work

Archives

July 2010
June 2010
May 2010
March 2010
February 2010
January 2010
November 2009
October 2009
September 2009
August 2009
July 2009
June 2009
May 2009
April 2009
March 2009
February 2009
January 2009
December 2008
November 2008
October 2008
August 2008
July 2008
June 2008
May 2008
April 2008

Recent Comments

admin on การเรียนรู้ด้วยวิธี linear least squares
Jirach on การเรียนรู้ด้วยวิธี linear least squares
Chotika on การเรียนรู้ด้วยวิธี linear least squares
admin on การเรียนรู้ด้วยวิธี linear least

```
doc2.add(new Field("title", "หัวเรื่องของเอกสารที่ 2", Field.Store.YES,
Field.Index.TOKENIZED));
doc2.add(new Field("content", "ผมหวังว่าตัวอย่างการใช้งานอย่างง่ายๆนี้ จะเป็นประโยชน์แก่ผู้เริ่มต้น
ทุกท่านครับ วิทวัส จิตกฤตธรรม (ผู้แต่ง)", Field.Store.COMPRESS, Field.Index.TOKENIZED));
doc2.add(new Field("url", "http://www3.anothersampledoc.co.th/doc2.html",
Field.Store.YES, Field.Index.NO));

/**Add the documents to your indexwriter*/
writer.addDocument(doc1);
writer.addDocument(doc2);
}
public static void main(String[] args) throws Exception {

////////////////////////////////////////////////////////////////////////
////////////////// BEGIN INDEXING PART//////////////////////////////////
////////////////////////////////////////////////////////////////////////

/**Let's build the index first.*/
/**The directory path to store index. THis must be a directory (folder).
* Modify to an appropriate location on your machine.
*/
String indexPath = "/home/nook/Desktop/demo_index";

/**Create a Lucene directory.*/
Directory dir = FSDirectory.getDirectory(indexPath);

/**An analyzer is used to analyze and tokenize the text added
to the index. For Thai, use ThaiAnalyzer. You can find it at

http://sansarn.com/look/download.html*/

Analyzer analyzer =
new ThaiAnalyzer();
//                new StandardAnalyzer(); This is for English

/**Construct an IndexWriter using the index directory and the analyzer.
The third boolean argument is to specify whether you want to create
a new index. In this case (true), we will create a new index. If the old
index exists in the directory, it will be OVERWRITTEN.*/
IndexWriter writer = new IndexWriter(dir, analyzer, true);

/**Add some documents to the index so that we can search.*/
addSampleDocuments(writer);

/**We optimize the index so that we can search faster.
* During the indexing, many files (index segments) will be generated.
THis optimization tries to reduces the number of these segment files.*/
writer.optimize();
/**DON"T FORGET TO CLOSE THE IndexWriter*/
writer.close();

////////////////////////////////////////////////////////////////////////
////////////////// BEGINE SEARCH PART //////////////////////////////////
////////////////////////////////////////////////////////////////////////
/**Pass the index directory to construct an IndexSearcher*/
IndexSearcher searcher = new IndexSearcher(dir);

/**In Lucene, query is modeled by a Query class. If you are a beginning and don't
want
* to construct the Query object by yourself, you can use QueryParser class.
QueryParser
* will construct a Query object from the query string that you input.
*/
QueryParser qParser = new QueryParser(
"content", //this is the default field to search.
analyzer //The analyzer is needed to tokenize your keywords. For example
//You search โรงเรียนต่างจังหวัด. The analyzer may tokenize the word into
//โรงเรียน and ต่างจังหวัด so that the search can be more flexible.
);

/**Your query*/
String queryString = "ตัวอย่าง วิทวัส"; //Normal operator is OR. = ตัวอย่าง OR วิทวัส
/**Parse the query and construct a Query object*/
Query luceneQuery = qParser.parse(queryString);

/**Search using the luceneQuery. In Lucene, search results are contained in Hits
object.
One Hit means one hit search result. So, Hits is an object containing a collection
of hit
search results.*/
Hits hits = searcher.search(luceneQuery);

for(int i=0;i
/**This is the document hit by the query*/
Document docI = hits.doc(i);
/**Get the url field*/
String url = docI.get("url");
/**Get the title field*/
String title = docI.get("title");
/**Get the content field*/
String content = docI.get("content");

/**Show all fields to the user*/
System.out.println("URL: "+url);
System.out.println("Title: "+title);
System.out.println("Content: "+content);
System.out.println("------------------------------");

}

/**Don't forget to close the searcher when you are done.*/
searcher.close();

/**Try to change to query to see different results.*/
}
}
```

หวังว่าคงมีประโยชน์แก่ผู้เริ่มต้น 😃

Tags: java, lucene

squares

Ake on การเรียนรู้ด้วยวิธี linear least squares

License

11 responses so far

จั่น May 9, 2009 at 12:00 am

เขียนได้ชัดเจน มีคำอธิบายชัดมาก รวมไปถึง code เขียนแล้วอ่านง่าย มีศิลปะ ขอบคุณมากๆครับ

จั่น May 9, 2009 at 12:10 am

สงสัยตรงที่

QueryParser qParser = new QueryParser(
“content”,
analyzer
);

แบบนี้ในการค้นหา เราจะสามารถค้นหาจาก field “content” เท่านั้น ใช่ไหมครับ (จากตัวอย่างนี้ยังมีอีก field หนึ่งที่สามารถใช้ค้นหาได้คือ field “title”) เราจะเขียนอย่างไรให้สามารถค้นหาได้ๆ หลายๆ field หรือว่า มีข้อจำกัดที่ค้นหาได้เพียง field เดียวหรือเปล่าครับ

จั่น May 9, 2009 at 1:43 am

ได้แล้วครับ ใช้ Class MultiFieldQueryParser ซึ่ง extends Class QueryParser

MultiFieldQueryParser mqParser = new MultiFieldQueryParser(new String[]{“title”, “content”}, analyzer);

แหล่งข้อมูล
==========
http://www.netlikon.de/docs/javadoc-lucene/lucene_2_3/org/apache/lucene/queryParser/MultiFieldQueryParser.html

จั่น May 9, 2009 at 2:02 am

สงสัยต่อครับ ตรง

doc1.add(new Field(“title”,”Test Lucene.”,Field.Store.YES));

เวลาผมค้นด้วยคำสั่ง
Query luceneQuery = qParser.parse(“Lucene”);

จะไม่พบข้อมูลครับ

แต่ถ้าที่ doc1.add (XXX) นั้น เปลี่ยนเป็น

doc1.add(new Field(“title”,”Test Lucene”,Field.Store.YES));

(เอา . หลัง Lucene ออก)

แล้วใช้คำสั่งเดิม

Query luceneQuery = qParser.parse(“Lucene”);

จะหาข้อมูลพบครับ

“.” มีผลอะไรกับการค้นหาครับ งงๆ???

admin May 9, 2009 at 2:20 am

ขอบคุณที่เข้ามาอ่านครับ

ตอบเรื่องหาหลายๆ field ครับ: ใช้ MultiFieldQueryParser ก็เป็นวิธีหนึ่งที่ทำให้หาหลายๆ field ได้ครับ แต่ก็มีวิธีอื่นอีกครับ เช่น สร้าง object Query ขึ้นมาเองเลย อาจจะใช้ BooleanQuery ให้แต่ละ clause เป็น TermQuery แล้วเชื่อมด้วย or/add ตามสะดวก จริงๆแล้วถ้าเราสร้างคำขอผ่าน

QueryParser หลังจากมัน parse ค่าขอที่เป็น String ที่เราป้อนให้แล้ว มันก็จะสร้าง object ของ class Query อยู่ดีครับ ฉะนั้นถ้าเรารู้แน่แล้วว่าจะใช้ Query แบบไหน (อาจจะเป็นแบบ TermQuery, PrefixQuery, ...หรืออื่นๆ) เราก็สร้าง object Query ตรงๆเลยก็ได้ครับ ไม่ต้องใช้ parser ก็ได้ แต่ในตัวอย่างใช้ QueryParser เพราะคิดว่าน่าจะเข้าใจง่ายครับ

ลองดู Query แบบต่างๆใน Lucene ได้ที่ http://lucene.apache.org/java/2_3_2/api/org/apache/lucene/search/Query.html ครับ

ตอบเรื่อง “.” เกี่ยวอะไรกับการค้น ? :

จริงๆแล้วไม่น่าเกี่ยวครับ ผมเดาว่า Analyzer ที่ใช้อยู่ตัดคำผิดครับ นั่นคือเนื้อหาที่มีคือ “Test Lucene.” ตัว Analyzer ที่ใช้อยู่มันอาจจะเข้าใจผิดตัดคำเป็น “Test” กับ “Lucene.” ที่นี้เวลาหาเราหา “Lucene” ซึ่งไม่เหมือนกับคำว่า “Lucene.” มันเลยไม่พบครับ ได้ลองหา “Lucene.” รึยังครับ ถ้ามันหาเจอแปลว่ามันตัดคำผิดครับ

jj September 20, 2009 at 8:16 pm

ขอบคุณนะคะที่มีบทความดี ๆ มาให้อ่าน

ตอนนี้ทำโปรเจคอยู่ค่ะแล้วต้องใช้ thaiAnalyzer ในการตัดคำ

แล้วนำคำที่ thaiAnalyzer ตัดแล้วไปใช้งานต่อ

แต่มีปัญหาน่ะค่ะไม่รู้จะดึงคำที่ thaiAnalyzer ตัดมาใช้งานได้อย่างไร

ช่วยตอบหน่อยนะค่ะตอนนี้ติดปัญหานี้ทำต่อไม่ได้เลย

admin September 21, 2009 at 10:16 am

จริงๆแล้วมีวิธีเอาคำที่ Analyzer ตัดมาใช้ครับ แต่ว่าถ้าต้องการตัดคำแต่ไม่ต้องการใช้ Lucene ผมว่าใช้ตัวตัดคำอันอื่นดีกว่าครับ ThaiAnalyzer ที่แนะนำไปข้างในคือตัดคำโดยใช้ดิกครับ ถ้าต้องการโปรแกรมตัดคำโดยใช้ดิกลองดูที่ http://www.sansarn.com/lexto/ ครับ เป็น Java เหมือนกันพัฒนาโดยทีมเดียวกันครับ

หรืออีกตัวเป็นตัวตัดคำที่ดีกว่าใช้ Conditional Random Field มาช่วย http://www.sansarn.com/tlexs/ โดยทีมสรรสารเช่นกันครับ

หากต้องการเอาคำจาก Analyzer จริงๆ เดี๋ยวผมดูให้อีกทีครับ ตอนนี้ยังไม่ค่อยสะดวกครับ

jj September 25, 2009 at 2:12 am

ต้องการนำคำที่ตัดจาก Analyzer ไปใช้ในโปรเจคจริง ๆ ค่ะ

รบกวนด้วยนะคะ

ขอบคุณนะค่ะ

admin September 26, 2009 at 5:39 pm

ตอบไว้ให้ที่ post ใหม่ครับ http://wittawat.com/blog/?p=284

กาย February 16, 2010 at 4:22 pm

เรียนถามครับว่า มีโปรแกรมสำเร็จรูป ตัดคำไทย ระดับ word และ character N-Gram ไหมครับ เหมือนเนคเทค ทำlexto กับ tlexs ให้เอาไปใช้
จะไปลองตัดคำ ประเภทข่าวดู

ขอบคุณครับ

admin February 16, 2010 at 5:12 pm

ไม่แน่ใจว่าผมเข้าใจถูกมั้ย แต่ว่าถ้าจะหาโปรแกรมตัดคำระดับ word ก็คือ lexto กับ tlexs อยู่แล้วนี่

ครับ

ส่วน character N-gram อันนี้ผมไม่ค่อยเข้าใจครับ ยกตัวอย่างได้มั้ยครับ ถ้าอยากได้แบบ เช่น
input: สวัสดี
output ที่อยากได้: สว, วั, ัส, สด, ดี

แบบนี่หรอครับ ถ้าแบบนั้นผมคิดว่าเขียนเองได้ไม่ยากมากครับ

Leave a Reply

Name (required)

Email (required)

Website

Submit Comment