Concordia University
Department of Computer Science and Software Engineering

Final Project

By
Maria Akther
ID: 27133596

Nadia Bilal
ID: 40009703

Ali Sangari
ID: 6816304

COMP6791: Information Retrieval and Web Search
Fall 2015
Date Submitted: December 07, 2015

ABSTRACT

Information retrieval is one of important activity in modern world. It involves techniques on building, representing, searching and manipulating large amount of data that provides services to retrieve information ones look for within shortest time and cost.

In this report, we discuss the approaches to perform a computational sentimental analysis. The major concern is to know about how a web crawler works to collect the web documents for building a searchable index. As well as, explore the sentiment for each of the document using the sentiment dictionary aFinn and evaluate these documents based on three way classifier into positive, negative, and neutral.

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 OBJECTIVES

The objectives of this project are to crawl the web links for a given set of root web pages, implement and prepare the test set of HTML files for testing and develop a system to analyze sentiment score for each document. As well as find out advantages and disadvantages of the overall systems.

1.2 BRIEF DESCRIPTION

1.2.1 CRAWLING

The definition for web crawler as stated in "An Introduction To Information Retrieval" by Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze is that [1]- "Web crawling is the process by which we gather pages from the Web, in order to index them and support a search engine." Sometimes a web crawler is referred to as robot, spider or a web scutter etc. It facilitates web users to crawl over web pages automatically. Here, we've used a java class library named WebSPHINX, an open source crawler which comprises two parts- crawler workbench and WebSPHINX class library. [2]

1.2.2 INDEXING

An inverted index is an index data structure storing a mapping from content, such as words or numbers, to its locations in a database file, or in a document or a set of documents. Inverted Index has proved to be the most efficient data structure for large-scale text retrieval systems [7]. Instead of searching the text directly into the documents, it searches an inverted index. For every term that appears in a given text collection, the inverted index contains a list of all positions in the text at which the term occurs. Efficient construction of inverted indexes is essential for large collections of text data. There are many kinds of text search library available for indexing and searching such as Lucene, Solr, Whoosh, swish-e etc.

SPIMI has been implemented to generate separate dictionaries for documents of each department without maintaining term-termID mapping across blocks and accumulate postings in postings lists. Then with these two ideas a complete inverted index can be generated for each block. Finally these separate indexes can then be merged into one big index. The most important advantages of SPIMI are to reduce the cost of disk accesses during inversion of large volumes of data and therefore there is less usage of disk space. [5][6]

1.2.3 SENTIMENT ANALYSIS

Sentiment analysis can be defined as a method to extract the behavior and to categorize the sentiment of a text unit. [4] The other name of it is “opinion mining”. Now a days, most of the companies benefit from it to make better their businesses. Sentiment analysis can be utilized for ecommerce product reviews, social media analysis as well. We’ve used the sentiment dictionary AFINN-111, the newest version with 2477 words and phrases to measure the sentiment for documents of each ENCS department from Concordia University website. Depending on the sentiment score, we can rank and compare the popularity of different ENCS departments. In general, it’s not guaranteed that the sentiment evaluation will be as accurate as human analysis.

2.0 DESIGN AND IMPLEMENTATION

In this section, the design of the Search Engine is described that includes the test environment such as which crawler, tools and techniques have been used to implement the system.

2.1 DEVELOPMENT AND TEST ENVIRONMENT

Title	**Description**
Crawler	Websphinx
Language	Python,Java
IDE	Jet Brains PyCharm 9.0, Eclipse
Sentiment analysis Library	aFinn

2.2 DESIGN DIAGRAM

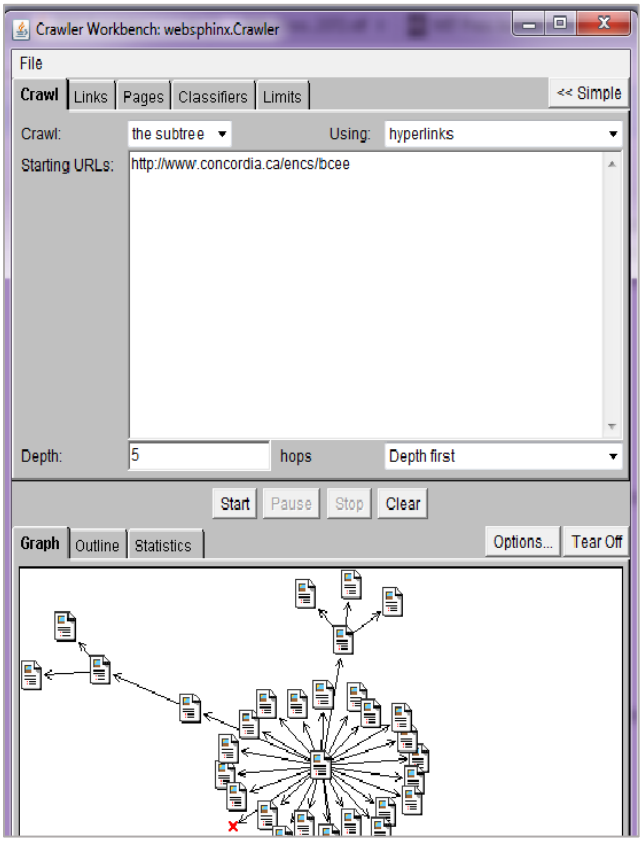

Fig: Crawler

Following is the design diagram of application which created using online tool, www.gliffy.com:

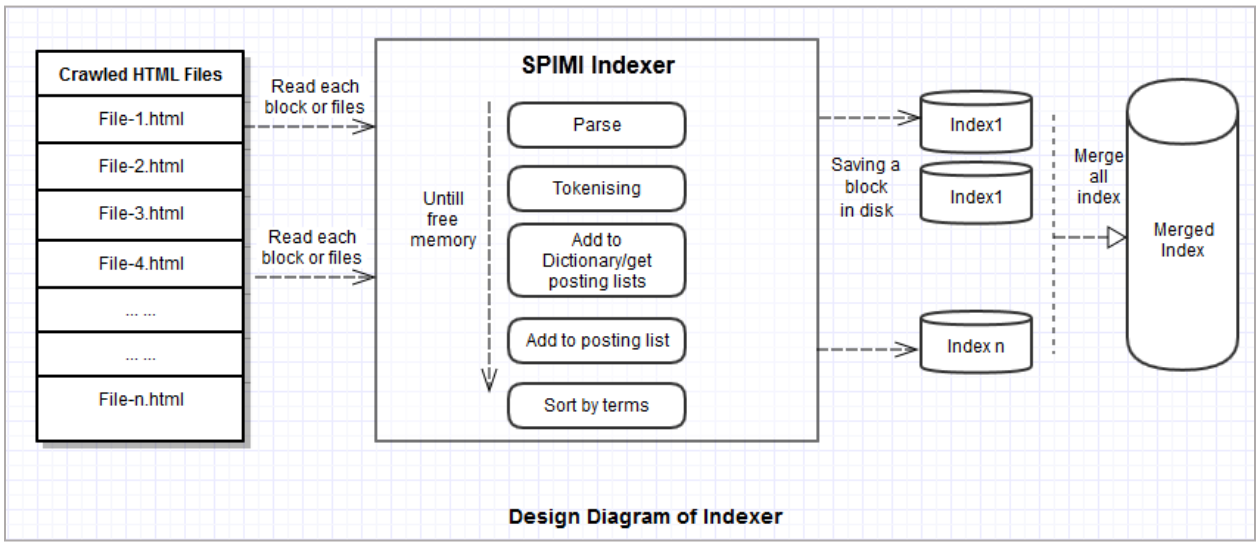

Fig. 1: Design Diagram of Indexer

2.3 IMPLEMENTATION

Departments	Number of Terms	Number of Documents/html pages
Bcee	1939	28
computer-science-software-engineering	1965	26
electrical-computer	2489	26
eng-society	867	7
info-systems-eng	2186	25
mechanical-industrial	1262	21
mystery-pages	586	9

Table: Size of Indexes

3.0 TESTING AND RESULTS

3.1 TESTING DATA SET

In our project we used the data collection for each department of ENCS (Faculty of Engineering & Computer Science) of Concordia University website those we crawled before using WebSPHINX.

3.2 RESULT

From the sentiment score, we can easily classify each department as positive, negative or neutral.

Departments	Sentiment Score
bcee	1601
computer-science-software-engineering	756
electrical-computer	1514
eng-society	349

info-systems-eng	1464
mechanical-industrial	1038
mystery-pages	451

3.3 FINDINGS

We have evaluated the sentiment for each ENCS department based on the crawled web documents and from the outcome, now we can answer for the following questions:

A. Which is the most positive Department in ENCS at Concordia?

We found the most positive department is bcee (Department of Building, Civil & Environmental Engineering) with highest score 1601.

B. Is Computer Science and Software Engineering more positive or less positive than Electrical and Computer Engineering?

According to the sentiment score, Computer Science and Software Engineering is **less positive** than Electrical and Computer Engineering.

C. Rank the departments in ENCS by sentiment of their web documents.

Total sentiment score for "bcee" is 1601

Total sentiment score for "electrical-computer" is 1514

Total sentiment score for "info-systems-eng" is 1464

Total sentiment score for "mechanical-industrial" is 1038

Total sentiment score for "computer-science-software-engineering" is 756

Total sentiment score for "eng-society" is 349

Total sentiment score for "mystery pages" is 451

D. Classify the departments in ENCS with a three way classifier into positive, negative, and neutral.

Based on the values that we have from the Sentiment analysis for each department the following as Positive and Negative threshold values and between the thresholds was considered a Neutral zone.

Lower Bound: 1000

Upper Bound: 1500

Positive	Negative	Neutral
electrical-computer	eng-society	mechanical-industrial
Bcee	computer-science-software-engineering	info-systems-eng

E. Classify the additional mystery page results the same way and compare its dominant sentiment.

We had some mystery pages in our data set those can be classified as ***negative*** according to the sentiment score.

F. What was the hardest step?

Sentiment analysis is appeared to be the hardest step during the project.

G. How big is the index?

The index contains of total 11294 terms for 142 documents of 6 ENCS departments and other relevant information.

4.0 CONCLUSION

Sentiment analysis is heavily applied in customer care and customer satisfaction programs of many organizations. Normally teams involved in these activities (above mentioned activities), classify and group their source documents through specific tools and manual research and then fed the resulting categorized documents into their Sentiment analysis engine. But in NLP research, Sentiment analysis could be tied with a document categorization/classification algorithm to first group the related* documents together and then run sentiment analysis on these groups' sentiment values and then intelligently and autonomously define thresholds for a three way classifier.

Document relativity would be clearly defined for each project and most likely would vary from project to project. We learned about several open source projects, mainly sponsored by

universities and research bodies that try to tackle different aspects of Natural Language Processing (also sometimes referred to as free-format text processing).

NLP research is concerned with artificial intelligence (AI) algorithms for machine learning to solve advance classification problems. Because NLP software solutions are very resource intensive with sometimes massive time complexity, another challenge in NLP is optimization of text compression and search methods and algorithms. Technology advancements in areas such as distributed computing and Big Data help scientific research and development NLP solutions.

4.1 ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty for his divine blessing makes us possible to complete this project successfully. We fell grateful to and wish my indebtedness to Dr. Sabine Bergler, Professor, Department of Computer Science, and Concordia University. Deep knowledge & keen interest of our instructor in the field of web information retrieval influenced us to carry out this project. Her scholarly guidance, continual encouragement, constructive criticism at all stage has made it possible to complete this project.

We would like to thank our all TA of this course. Their endless patience, valuable advice, reading many inferior drafts and correcting them helped a lot. We would also like to express our heartiest gratitude to our friends who helped us by sharing their knowledge and experiences.

Finally, we must acknowledge with due respect the constant support and patients of our families.

REFERENCES

[1] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze “An Introduction to Information Retrieval”, 9th Edition.

[2] https://www.cs.cmu.edu/~rcm/websphinx/

[3] http://www.concordia.ca/encs.html ;

[4] http://lct-master.org/files/MullenSentimentCourseSlides.pdf

[5] Efficient Single-Pass Index Construction for Text Databases by Steffen Heinz and Justin

Zobel School of Computer Science and Information Technology, RMIT University, GPO Box 2476V, Melbourne 3001, Australia

[6] Memory Management Strategies for Single Pass Index Construction in Text Retrieval Systems by Stefan Buttcher and Charles L. A. Clarke, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada

[7] https://www.elastic.co/guide/en/elasticsearch/guide/current/pluggable-similarites.html

[8] Introduction to Sentiment Analysis, http://www.slideshare.net/makrandp/introduction-27376010

[9] http://nlp.stanford.edu/sentiment/

[10] http://stackoverflow.com/questions/6073109/sentiment-analysis-api-tool-for-java

[11] Code References: https://code.google.com/p/website-clustering/source/browse/trunk/src/ by Marc-Andre Faucher