RMIT University, School of Computer Science and Information Technology
COSC 2305/2306 Search Technology

ASSIGNMENT 3

(Worth 60% of the total assessment for this course)

Aim

The aim of this assignment is to investigate, analyse, design, and implement an algorithm for automatic keyphrase extraction from abstracts of scientific articles using MapReduce.

*This is an individual assignment and you are required to submit **original** work, any work copied or derived from another source must be identified and details provided in your source files and in your written submission.*

Description

Keyphrases are words that capture the main topic of the document. As they represent the key ideas of the documents, extracting good keyphrases benefits various natural language processing (NLP) applications such as summarization, information retrieval (IR) and question-answering (QA). In summarization, the keyphrases can be used as a semantic metadata [1,2,3]. In search engine such as Google and Yahoo, the keyphrases play a role to supplement full-text indexing and to assist users to create good queries. Therefore, having good keyphrases impact on the quality of NLP applications.

In the project, you will be provided with set of abstracts of scientific articles (i.e. papers) and will be asked to produce the keyphrases for each article.

Task

Three tasks

(1) preprocessing such as Part-of-Speech tagging, lemmatization and/or stemming (optional)

(2) candidate generation

(3) candidate ranking

You are required to make appropriate use MapReduce to produce a solution that is scalable to very large amounts of data.

Data

The total number of abstracts is 500. 250 abstracts will be given as trial data and another 250 abstracts will be given for test data on later in the semester which students need to extract keywords. Abstracts are named as "number.abstr" and contain the title and the abstract. The answers for each abstract are provided for trial data in the file, "trial.answer". The format of answers in "trial.answer" is "FILENAME\s:\sKEYPHRASE_LIST" separately by a comma where "\s" means a space.

e.g. "4 : computer science,agent system,multi-agent"

Data statistics

On average, the abstract has 9.824 keywords and 226 abstracts have at least 10 keywords.
The maximum number of keywords in an abstract is 31 which occurs once.

Format of your answer file

For given 250 test data, you need to assign up to 10 keywords for each abstract.

To submit answer files, please ensure that you follow the below conventions.

- "FILENAME\s:\sKEYPHRASE_LIST" separately by a comma where "\s" means a space.
- Please list 10 candidate keyphrases per document. Your submission's score will be based only on the first 10 answers.
- We accept one type of keyphrase format.
- Stemmed words using "Porter stemmer" (porter.pl) written in Perl which we will provide for the project. To use "porter.pl", "./porter.pl input > output"
- Output file of stemmer contains stems corresponding to words in the same line in input file, e.g. with document C-1

stemmed words = C-1 : keyphras,extract,competit,test,perform

- You can submit up to three different types of output.

The output file name should start with your student ID.

For example, if your student ID is 123456, then your file name starts with "123456.out".

If you provide 3 outputs, the file names are "123456.out-1", "123456.out-2" and "123456.out-3".

Performance

The performance will be measure by micro-average Precision, Recall and F-score based on your submission(s).

The script to measure performances is "performance.pl" (written in Perl). The script compares the stemmed keyphrases between your answer set and the provided answer set.

- usage: perl performance.pl <your_file_name>

(e.g. "perl performance.pl 123456.out")

Components of assessment

This assignment will be marked out of 60. The assignment contributes 60% towards your final mark for the course. It comprises the following deliverables.

(a) Implementation (**due**: 11.59pm Thursday 30th September2010) ***40 marks***

You are required to submit your source code, evidence of having running your program (e.g. diagnostics produced by MapReduce, together with output files for up to three runs.

(b) Brief presentation (5 minutes) in class (Tuesday 12th October 2010) ***5 marks***

You will present to the rest of the class, how your program worked with an emphasis on any features you considered to be novel, and how you made use of MapReduce.

(e) Written report on project (**due**: 11.59pm Friday 15th October 2010) ***15 marks***

- You are required to write a 2 to 4 page report in the style of a research paper (to be submitted as a PDF file) on your program, describing your approach, implementation choices and both how it performed in effectively finding keyphrases, and how efficient it was including how scalable your solution is. Your report should also include list or sources you have used.
- The report format similar to a research paper submitted to Conferences in Research and Practice in Information Technology, you may use either the Word or LaTeX template available at: http://crpit.com/AuthorsSubmitting.html
- A PDF file of your report should be emailed to the lecturer james.thom@rmit.edu.au within three days of the discussion (or as otherwise negotiated with the lecturer).

Late submissions

Late submission of assignments will be penalised as follows:
For assignments 1 to 5 days late, a penalty of 10% (of total available marks) per day.
For assignments more than 5 days late, a penalty of 100% will apply.

Further instructions

If clarification of any details of the assignment are required, these will be announced via blackboard on myRMIT.