Lab Course: Distributed Data Analytics
Exercise Sheet 7

Mofassir ul Islam Arif
Information Systems and Machine Learning Lab
University of Hildesheim
Submission deadline: Monday 29th June 2002 10:00AM (on LearnWeb, course code: 3116)

Instructions

Please following these instructions for solving and submitting the exercise sheet.

1. You should submit two items. a) a zip or a tar file containing python scripts and b) a pdf document.
2. In the pdf document you will explain your approach (i.e. how you solved a given problem), and present your results in the form of graphs and tables.
3. The submission should be made before the deadline, only through learnweb.

Exercise 1 (10 Points)

You need to fulfill all the requirements for this questions and include in you report, outputs from your solution i.e. performance analysis and convergence graphs by varying number of workers i.e. mappers and reducers. You should use Apache Hadoop MapReduce for solving this problem.

Problem statement

You are familiar with solving recommender systems problem using a matrix factorization. In this task you have to parallelize coordinate descent method for matrix factorization using MapReduce framework. Sample parallel coordinate descent algorithms are explain in resources mentioned in "Related reading material". The objective function optimizes a square loss.
You should be able to present your solution with the following information (not limited to).

1. Data division strategy?
2. How you parallelize your algorithm? Explain Map and Reduce functions
3. Any other optimization technique you used to optimize MapReduce i.e. caching.
4. Present performance analysis i.e. speedup graph, [hint: have a look at the "Related reading material" how to calculate speedup].
5. Measure the affect of using varying number of workers on the convergence].

Dataset

For this exercise you will use movielens10m or movielens20m dataset available at `http://files.grouplens.org/datasets/movielens`. The movielens dataset is a rating prediction dataset with five star ratings (on a scale of 1-5). [Hint: you can start with a smaller movielens dataset i.e. movielens1m or movielens100k. But your final solution should be based on 10m or 20m dataset]

Related reading material

1. Parallel Coordinate Descent `http://www.caam.rice.edu/~optimization/L1/optseminar/Parallel%20BCD.pdf`
2. Matrix factorization with coordinate descent `http://www.cs.utexas.edu/~cjhsieh/icdm-pmf.pdf`
3. Parallel Speedup analysis `https://portal.tacc.utexas.edu/c/document_library/get_file?uuid=e05d457a-0fbf-424b-87ce-c96fc0077099&groupId=13601`

4. Spark launching configuration https://spark.apache.org/docs/latest/submitting-applications.html

1. Hadoop Linux Setup urlhttps://phoenixnap.com/kb/install-hadoop-ubuntu
2. Introduction to Natural Language Processing (NLP): http://blog.algorithmia.com/introduction-natural-language-processing-nlp/
3. TFIDF http://www.tfidf.com/ and https://en.wikipedia.org/wiki/Tf%E2%80%93idf
4. Tutorial: Finding Important Words in Text Using TF-IDF http://stevenloria.com/finding-important-words-in-a-document-using-tf-idf/
5. Common English stopwords: http://www.textfixer.com/tutorials/common-english-words.txt
6. Data-Intensive Text Processing with MapReduce http://www.umiacs.umd.edu/~jimmylin/MapReduce-book-final.pdf