CSE 587 Data Intensive Computing Spring 2019

REPORT - LAB 2

Author : HRISHIKESH NITTURKAR

Submitted on: 04/21/2019

Introduction:

In this project, the goal is to gather data from different sources for various topics and process it using the Hadoop File System. The whole project revolves around various components of the Hadoop File System such as the mappers, reducers etc. We use these components and run algorithms such as Word Count and Word Occurence and observe how the data is processed by the mappers and how the output is produced by the reducers. We know that in a Hadoop file system, the mappers take the input from the user in huge quantities and process it sequentially and parallely. There are many parallel mappers working continuously and parsing the data according to the algorithm we provide. The output from all the mappers in stored or held in the threshold barrier until all the mappers process every last bit of data. Then this output from mappers is served as input for reducers at once which process the data parallely and synchronously and finally the output is generated.

Description:

Data Collection:

Firstly, the topic I chose is “sports” and the subtopics are “tennis”, ”basketball”, ”golf”, ”baseball”, ”soccer”. The programming language I chose is Python. I started by collecting the twitter data first. I collected more than 20000 tweets for the topic and subtopics combined. For the New York Times data collection, I registered for an API key and downloaded 100 articles for topic and subtopics each. For common crawl data extraction, I used indexed files from common crawl archive and retrieved different records from the warc files downloaded from the indexed files. Then I retrieved all the urls from the records and filtered the urls to get the data needed. I got 100 articles for each of the topics and subtopics.

Data Cleaning:

All the data was processed and cleaned thoroughly using regex, nltk library, stop words etc. Preprocessing includes cleaning the special characters, urls, unnecessary words such as stop words which are a, an, are, the etc. After cleaning the data, each article’s data is stored in a separate file under the topics name which is under the data source used. For example all data collected from New York Times is stored under the directory “NYData” which has subdirectories relating to topic and subtopics which had all the articles in separate files. This file structure is followed for the remaining data sources. In the parent folder, there are 3 folders - one for each data source. In each of these folders, we have folders for topic and subtopics which store the articles pertaining to that topic. In addition to these files, there is a mapper file for Word Count and Word Cooccurrence algorithms, a reducer file for Word Count and Word Cooccurrence algorithms, two tableau workbooks

and word cloud images for the output provided for all the articles in the data source. In addition to these folders, there are folders that store the output from the hadoop filesystem for word count with all words(), word count that returns only the top 10 repeated words and the word co-occurrence algorithm. Graphically, the file system can be represented as

Hnitturk(*directory*)
- |_CCData(*directory*)
  - |_baseball
  - |_sports
  - |_basketball
  - |_soccer
  - |_golf
  - |_tennis
  - |_output_top_10_words
  - |_output_all_words
  - |_output_word_co_occurence
  - |_*mapper.py*
  - |_*reducer.py*
  - |_*woc_mapper.py*
  - |_*woc_reducer.py*
- |_NYData
- |_twitterData
- |_demo
- |_Images
- |_*mapper.py*
- |_*reducer.py*
- |_*woc_mapper.py*
- |_*woc_reducer.py*
- |_*WebPage.html*

The above directory listing shows the listing for one folder which is just repeated for the other folders.

Setting up Hadoop Infrastructure:

I have used the cloudera hadoop infrastructure details provided and set up the file system. The preprocessed data is fed into the Hadoop File System and links to the mapper and reducer are also provided so that the hadoop file system can run the algorithms on the data provided. From the Hadoop file system, we get the output in the form of a folder with a success message file and the output file. We get this data to our local file system and use that to do visualizations and learn about trends and topics that are of interest to people.

Algorithms:

In this project, I have used two main algorithms, Word Count and Word Co-occurrence. A copy of code for these algorithms is stored in each of the data source folders. These algorithms are run on the total data from each of the sources in the HDFS. For example, the algorithms are run for all the ~600 articles obtained from New York Times and Common Crawl Data and the tweets collected from Twitter. The Word Count algorithm counts the frequency of all the words in the articles and the Word Occurrence algorithm gives the occurrence of the top 10 words from the Word Count algorithm with the next word in the article.

Visualization:

I used Tableau to create the word clouds for the top 10 words with highest count from the Word count algorithm and also the top 50 pairs from the output of Word co-occurrence algorithm to make sure the data is readable. All the word cloud images are saved in the respective data source folders along with the Tableau workbooks. In addition to this, I have also designed a HTML page which can help compare the word clouds for various data side by side and also the word co occurrence data.

The above process can be used extensively for figuring out the trends in various fields, also other than sports such as medicine, politics etc. These trends are very important to monitor from a business point of view. This type of analysis can be used to better understand how to tweak a product so that the public receives it better. Also, this type of analysis is used to predict the trends based on the news articles and tweets posted by people. The hadoop file system provides the much needed parallelism in handling the huge amounts of data that the data mining can produce. It was a really interesting project to work on and figure out how to make optimise the hadoop speed, filter out data to make the output more credible and reliable.