Programming Assignment 1

Creating N-gram profile for a Wikipedia Corpus

Due: Feb. 21, 2018 5:00PM
Submission: via Canvas, individual submission

Objectives
The goal of this programming assignment is to enable you to gain experience in:

- Basic features of Hadoop distributed file system and MapReduce
- Creating NGram profiles using Hadoop MapReduce

1. Introduction

An N-gram is a contiguous sequence of N items from a given sequence of text or speech. An N-gram model is a type of probabilistic language model for predicting the next item in such a sequence in the form of a (n-1). N-gram models are widely used in statistical natural language processing. In speech recognition, phonemes and sequence of phonemes are modeled using a N-gram distribution. For sequences of words, the 1-grams (aka unigram) generated from "We analyze large dataset" are ("We", "analyze", "large", "dataset"). For the same sentence, the 2-grams (aka bigram) are ("__ , We", "We, analyze", "analyze, large", "large, dataset", "dataset, __"). Here, "__" represents the empty space before and after the sentence.

N-grams are used for various applications such as approximate matching, plagiarism detection, searching for the similar documents, automatic authorship detection, and linguistic cultural trend analysis. Google's Ngram Viewer is a good example of N-gram analysis[1].
(https://books.google.com/ngrams/info)

In this assignment, you will create N-gram profile of the corpus of selected Wikipedia articles[2]. You will: (1) extract all the unigrams, (2) compute the frequency of each unigram per page and also over the corpus, and (3) rank the unigram based on these frequencies.

As a corpus for this assignment, you will be provided around 1GB of dataset selected from Wikipedia articles. Your computing environment will be MapReduce. Installing/configuring Hadoop should have already been completed as part of PA0.

[1] https://books.google.com/ngrams/info

[2] https://dumps.wikimedia.org

2. Programming Requirements

2.1 Programming Language

You are required to use Java (Version 1.7 or higher) for this assignment.

2.2 Hadoop Configuration

For this assignment, you will be working on your own Hadoop Cluster, which should have finished this as a part of PA0. The walkthrough guidelines are available at:
http://www.cs.colostate.edu/~cs435/datafiles/PA0/HadoopInstallationGuide.pdf

3. Generating N-gram profile

In this assignment, you should generate unigram profiles. To extract unigram, you should tokenize the sentences based on the whitespace characters. Ignore tense, gender, and hyphenated words. "He" and "She" should be considered as the different words. "have" and "has", or "have" and "had" should be considered as the different words. "well-described" should be considered as 1 unigram. Please convert all of the upper cases to lower case. Also, consider only alphabetic and numeric text.

Your Ngrams should ignore any punctuations or apostrophes. Consider "you're" as a unigram. In your output, "you're" should be listed as "youre". The frequency of Ngram should count only the exact same appearance of text. Do NOT eliminate the stop words (e.g. "a", "the", or "are").

Your corpus will be a set of Wikipedia documents. Your software should generate the following Ngram profiles using MapReduce.

(1) Profile 1
A list of unigrams that occurred at least once in the entire corpus (1G dataset). The unigrams must be sorted in (ascending) alphabetical order. You should eliminate duplicates. The output should be generated using MapReduce. You may store the output in multiple files.

(2) Profile 2
A list of unigrams and their frequencies within the target article. Your software must generate this profile per article. Your list should be grouped by the Document ID (see page 3), and sorted (in descending order) on the frequency of the unigram within the article. This output should be the generated using MapReduce. Output may be stored in multiple files.

(3) Profile 3
A list of unigrams and their frequencies within the target corpus. The list of unigrams should be sorted (in descending order) on the frequency of the unigram within the corpus (1G dataset). This output should be generated using MapReduce. Output may be stored in multiple files.

The result of your computation should be stored as file(s). You should generate the result using Hadoop's MapReduce programing framework – there is a 100% deduction if you write a standalone program to do this.

3.1 Input data

Your input data will be a dataset compiled from the set of Wikipedia articles. You will be required to work with a ~1GB dataset consisting of files with Wikipedia articles.

The Input dataset is available at: [Link]

You will be required to submit the output of your software that processed this 1GB dataset.

Each data file is organized in the format described below.

```
…
Title_of_Article-1<====>DocumentID-1<====>Text_of_Article-1
NEWLINE
NEWLINE
Title_of_Article-2<====>DocumentID-2<====>Text_of_Article-2
…
```

There are 3 components describing an article: (1) Title of the article, (2) Document ID, and (3) Text of the article. The title (text information) represents the title of a Wikipedia article. The document ID is specified by Wikipedia and every article has a unique document id. Finally, the text of the article encapsulates what was included in that Wikipedia article. Each data file may contain multiple Wikipedia articles and these are separated by two consecutive `NEWLINE` characters.

```
April 28<====>1639<====>April 28 Events 224 – The Battle of Hormozdgān is
fought. Ardashir I defeats and kills Artabanus V effectively ending the
Parthian Empire.357 – Emperor Constantius II enters Rome for the first time
to celebrate his victory over Magnus Magnentius.1192 – Assassination of
Conrad of Montferrat (Conrad I), King of Jerusalem, in Tyre, two days after
his title to the throne is confirmed by election. The killing is carried out
by Hashshashin.1253 – Nichiren, a Japanese Buddhist monk, propounds Namu Myōhō
Renge Kyō for the very first time and declares it to be the essence of
Buddhism, in effect founding Nichiren Buddhism.

USA<====>453673<====>USA - The United States of America (USA), commonly known
as the United States (U.S.) or America, is a federal republic composed of 50
states, a federal district, five major self-governing territories, and
various possessions. At 3.8 million square miles (9.8 million km2) and with
over 325 million people, the United States is the world's third- or
fourth-largest country by total area and the third-most populous. The capital
is Washington, D.C., and the largest city by population is New York City.
Forty-eight states and the capital's federal district are contiguous and
located in North America between Canada and Mexico. The state of Alaska is in
the northwest corner of North America, bordered by Canada to the east and
across the Bering Strait from Russia to the west.
```

You will also find a smaller data file for testing out your software at [Link]. This file would be smaller than 10MB in size and is to be used for testing purposes only.

3.2 Output data

Each of the files should follow the file format below for the profile 1.

```
Ngram-A NEWLINE
Ngram-B NEWLINE
Ngram-C NEWLINE
…
```

Each of the files should follow the file format below for the profile 2.

```
Document-ID TAB ngram TAB frequency NEWLINE
```

Each of the files should follow the file format below for the profile 3.

```
ngram TAB frequency NEWLINE
```

For example, output of the profile 2 may contain lines;

```
…
453673     federal    3
453673     district   2
…
```

4. Submission

This assignment requires an **individual submission**. Please submit your tarball of source files and output data (From the full dataset) via Canvas. The source files should include your java code of Mapper/Reducer functions and any script file that you will used for demo. You will download your source files/script from Canvas for the demo. Do not miss any file. For your demo, you are not allowed to use any file outside of your Canvas submission.

5. Grading

Each of the submissions will be graded based on the demonstration of your software to GTA. During the demonstration, you should:

Step 1. Explain your approach and present the output from the 1GB dataset. (1 points)

Step 2. Execute your MapReduce software on a sample input data given to you during the demo of profile 1 and 2. (5 points)

Step 3. Go over the output files. (1 point)

Demo includes short interviews about your software design and implementation details. The interview questions may include a request for simple code or configuration modifications. Your inability to explain what you have done or to make simple modifications to the software you have written will impact your score. This assignment will account for 7% of your final grade. The grading will be done on a 7 point scale.

Deductions: If you implement this assignment using a standalone program you will have a 100% deduction i.e. your score for this assignment will be 0.

You are required to work alone on this assignment.

6. Late policy

Please check the late policy posted on the course web page.