COSC 589 - Web Search and Sense-making

Assignment 4
Due Wednesday March 14, 11:59pm

Task: Parse Wikipedia using Scala.xml

Introduction:

In this assignment, we will parse the Wikipedia dump using an xml parser and keep only the articles from the dump.

Requirements:

>100GB free disk space in your machine.

Instructions:

Write a WikiArticle.scala file to extract the articles from the English Wikipedia dump, by taking the following steps:

1. Read in the output file of your last assignment (one Wiki page per line).

2. Parse the file using the xml parser in scala. Make sure you can access the title and text fields.

3. Get the articles, one type of Wiki pages, from the pages. The Wikipedia page types are stubs, redirects, disambiguation pages, and articles. Articles are pages that are not stubs, redirects and disambiguation pages.

   You are welcome to use the following code template:

```
import scala.util.matching.Regex
import scala.xml.XML
import org.apache.spark.SparkConf
import org.apache.spark.SparkContext
import org.apache.spark.SparkContext._

object WikiArticle {
  def main(args: Array[String]) {
    val sparkConf = new SparkConf().setAppName("Wiki WordCount")
    val sc = new SparkContext(sparkConf)
    val txt = sc.textFile("./wikidump.page.per.line")  // your output from the last assignment

    val getTitleAndText = txt.map{ l =>
      val line = XML.loadString(l)
      val title = (line \ "title").text
      val text = ….  // You code come here, to get the strings in <text></text>
```

```
        // your code goes here to output the title and the text
      }

      // you will need to write a function isArticle
      val articles = getTitleAndText.filter { r => isArticle(r._2.toLowerCase) }

      // save the articles. See the format in 4.
    }
  }
```

4. Save the articles (using saveAsTextFile), in the following format:

   - Each line is one article
   - First element is the title of the article
   - Second element is the content of the article
   - The first and the second elements are delimited by the tab '\t'. Note that Spark's default delimiter is ','. We will have to use something different from it because lots of Wikipedia titles contain comma in themselves. To avoid the confusion, when we output the title and text, we separate them by a tab.

5. Print the total number of articles in English Wikipedia to the screen
6. Perform a WordCount for the Wikipedia Articles. Save the outputs.

What to Submit:

- Your code
- Screen capture of the article count results that you print to the screen
- Screen capture of the total number of words in Wiki Articles that you print to the screen
- Screen capture of the total number of unique words in Wiki Articles that you print to the screen
- Screen captures of the beginning of your saved articles (let us say the 2 screen captures of the first 20 lines on the screen)
- Screen captures of the beginning of your saved WordCount outputs (let us say the 2 screen captures of the the first 20 lines on the screen)

What NOT to Submit:

- Your input or output files

Where to submit:

- Canvas

When:

- Due on 3/14/2018, 11:59pm.