1. Overview

In this assignment you will build upon your first homework and develop ranking algorithms for your crawled Polar datasets. You will develop and compare two sets of ranking and retrieval approaches: content-based approaches that will leverage the indexed textual content of your data and summary metrics using IR techniques such as term frequency-inverse document frequency (TF-IDF) to generate relevancy; and link-based approaches that will leverage citation relationships (graph-based) between the indexed documents and information other than the textual content of the document to perform relevancy analysis.

The assignment will help you answer relevant scientific questions related to geospatial properties; measurements present in the file; citation and information source for the data; and trends and topics over time in the Arctic and Antarctic. The President’s National Strategy for the Arctic Region identifies the region as having critical natural and commercial resources including oil, iron and other ores, and national security interests (maritime/air/sea and land). However, the region is also recognized as highly impacted by climate change, especially e.g., in reduction of sea ice, caused by warming. Efforts are underway to aid in the responsible stewardship of the region, and your analyses and work...
will aid significantly in sifting and sorting through the vast number of results in Polar data archives.

2. Objective

The objective of this assignment is to develop a notion of relevancy related to the Polar datasets that you crawled and obtained in assignment #1. Please note, **You are not required to crawl for this assignment.** Instead, you will leverage the data that you obtained in assignment #1. If you do not have your data any more, please notify the Graders, TAs and Professor and be prepared to provide a reason as to why you don’t have it since the Professor has mentioned several times to date to keep your data.

As we have learned in class, relevancy can be measured in Information Retrieval via several methodologies; the first generation of relevancy was related to text-based summarization and we have discussed techniques such as TF-IDF as a metric in these means. With the advent of Hypertext Induced Topic Search (HITS) and more-so with the PageRank algorithm developed by Brin and Page, we have also seen a notion of link-based relevancy that does not rely on summarized text, but instead graph-based relationships between documents indicating their associated relevancy to one another. We will build on these two notions in this assignment.

You will deploy the Apache Solr ([http://lucene.apache.org/solr/](http://lucene.apache.org/solr/)) full text indexing system that is built on top of the Apache Lucene search engine ([http://lucene.apache.org/](http://lucene.apache.org/)). Solr and Lucene fully implement an inverted index and the vector space retrieval model, and provide a basis on which to develop relevancy mechanisms via text or otherwise. You will leverage Solr’s native integration with Apache Tika in this assignment via the ExtractingRequestHandler plugin in Solr. This allows you to directly post documents to Solr and run Tika on the server side to extract metadata and to index it. You will also leverage your Nutch crawl data in this assignment and Nutch’s direct integration with Solr via Nutch’s indexing plugin to use and compare your already extracted Tika-based metadata and text to that provided by Solr’s server side ExtractingRequestHandler.

While building your index you will make careful considerations based on the data model of your Polar data as to what fields are important for relevancy measurement. Is **timestamp** an important field? What about image metadata such as Color Component Quantization? In HTML pages, what metadata is available to you to relate to other documents? Do you have any videos and if so, how do the videos’ properties such as **Frames Per Second** relate them to other content that you have downloaded and indexed?

Beyond these data-centric questions, you will also work to develop a set of science-based questions that will test the boundaries of your text-based and link-based relevancy algorithms. To do so, you will engage in a Google Hangout with Dr. Annie Bryant Burgess who is doing her postdoc on information retrieval, Tika and Polar/Arctic data. Teams will have a session with Annie in which they develop key science questions that will drive relevancy measurements for your Polar datasets.
3. Tasks

1. Sign up for one of the three Google Hangout Sessions that Dr. Burgess will conduct (details to be provided by the Graders and TAs)
   a. Develop a set of “scientific questions” that can be answered related to the Polar data that you have captured.

2. Develop an indexing system using Apache Solr and its ExtractingRequestHandler ("SolrCell")
   a. Install Solr from the lucene_4_10 branch (instructions in Section 4)
   b. Upgrade SolrCell similarly to what you did in assignment #1
      i. Build Tika trunk (1.8-SNAPSHOT) with the following support
         ii. GDAL
         iii. OCR
         iv. FFMPEG
             1. Build Tika with FFMPEG support per [https://issues.apache.org/jira/browse/TIKA-1510](https://issues.apache.org/jira/browse/TIKA-1510)
   c. Your system should take, as input, the original Polar Data you crawled from assignment #1. Since you will be performing direct post of this data (without Nutch in this step), you must first using the ./bin/nutch dump command to export your data before posting it in to Solr.

3. Leverage the Nutch indexing system to build up an Apache Solr index
   a. Upgrade your Tika instance from assignment #1 to include FFMPEG support per: [https://issues.apache.org/jira/browse/TIKA-1510](https://issues.apache.org/jira/browse/TIKA-1510) and outlined in 2.iv above.
   b. Compare the metadata extracted from using Tika in Nutch during crawling upstream compared to your SolrCell based Tika run generated in Task #2.
      i. What did Tika extract in Nutch compared to what SolrCell extracts?

4. Design and implement two ranking algorithms for your Polar data documents
   a. A content-based algorithm that uses text retrieval methods including TFIDF to identify the relevancy of each document in your index. The algorithm should use the text stored in Solr fields (provided either by SolrCell and/or by Nutch and its use of Tika) to assess the relevancy of each document to a given query from your scientific queries set developed in Task #1.
   b. A link-based algorithm that uses the relationships between your associated documents (the metadata features) to identify relevancy of the documents independent of the user’s query. Link-based metrics compute relevancy based on properties such as geographic location (all images related to e.g., Barrow Alaska); temporal properties (all images and videos from
September 2015 or from 2014); and other relevant features. You are free to use the extracted features from Tika and/or to derive some of your own features as well. For example, you may want to have a look at the CLAVIN project for geographic topic identification

i. https://github.com/Berico-Technologies/CLAVIN

ii. You will need to develop a program in Python, and/or Java and/or Bash that properly augments your Solr documents with the tagged results of your link-based relevancy algorithm.

5. Develop a suite of queries that demonstrate answers to the relevant scientific questions that you came up with in Step #1. For illustrative purposes here are some examples of what these questions may look like

   a. What time-based trends exist for discussion of oil, iron and other natural resources in the Arctic region? Are documents and topics collocated to geographic region?

   b. How many regions of interest are represented by the data you collected? Identify geographic “regions” as e.g., Circumpolar Arctic region, Antarctica and the Southern Ocean. Can you use the distribution of your documents to represent territories?

   c. Can you predict areas in which there are national security interests (maritime/air/sea and land)? Which areas and why?

   d. Is there a trend with respect to science data and measurements related to Climate Change? Is it time-based and/or geographic region based? What areas show a high document relevancy for sea-ice extent and decline?

6. Develop a program in Java, Bash, and/or Python that runs your queries against your Solr index and outputs the results in an easy to read list of results demonstrating your relevancy algorithms and answers to your challenge questions from Task #5.

7. (Extra Credit) Develop a Lucene-latent Dirichlet allocation (LDA) technique for topic modeling on your index and use it to rank and return documents.

   a. Re-run your queries and examine the results from Task #4. What differences do you see? Can you explain them?

8. (Extra Credit) Figure out how to integrate your relevancy algorithms into Nutch.


9. (Extra Credit) Create a D3-based visualization of your link-based relevancy. Provide a capability to generate D3 relevancy visualizations as a Nutch REST service using Apache CXF.

4. Assignment Setup

4.1 Group Formation

Please keep the same groups as for your assignment #1. If you have any questions please contact:

Gouthami Kondakindi
Use subject: CS 572: Team Details

4.2 Dataset

Please start with your data that you have prepared in assignment #1. If you would like additional data from Amazon S3, please let the graders know and we will coordinate its delivery via amazon S3 buckets and read-only keys to one member of your group.

4.3 Installing and Building Apache Solr

You will need to build Apache Solr from the 4_10 branch of lucene-solr to take advantage of a fix that the Professor provided for integrating Tika with SolrCell and OCR:

http://svn.apache.org/repos/asf/lucene/dev/branches/lucene_solr_4_10/

You can find more information here:
https://issues.apache.org/jira/browse/SOLR-7137
https://issues.apache.org/jira/browse/SOLR-7139

Apache Solr comes with a web application server (Jetty), or you can also deploy and configure Solr with Apache Tomcat. Either way will work fine for this assignment and the instructions are provided here:

https://cwiki.apache.org/confluence/display/solr/Running+Solr+on+Jetty
https://cwiki.apache.org/confluence/display/solr/Running+Solr+on+Tomcat

You should also review the basic installation instructions:

http://wiki.apache.org/solr/SolrInstall

And the instructions for using SolrCell/ExtractingRequestHandler:

https://wiki.apache.org/solr/ExtractingRequestHandler

Once installed, you will need to configure Solr to accept your Polar data model. You will also be responsible for integrating your ranking algorithms into Solr and for using Solr to query and answer your challenge questions.

Please review Solr function query documentation:
http://wiki.apache.org/solr/FunctionQuery
Your ranking algorithms can either be interactive (per query), or also on document index-time ranking. This is something you will need to figure out as a group based on the type of ranking algorithm you are developing.

4.4 Upgrading Tika with OCR, GDAL and FFMPEG

Please follow similar instructions as described in assignment #1 and via the links above in Task #2b.

4.5 Indexing with Nutch

Please see: https://wiki.apache.org/nutch/bin/nutch%20solrindex for some guidance on how to index from Nutch into Solr. Please use your Nutch 1.10-trunk deployment from assignment #1 for this step.

4.6 Describing your Relevancy Algorithms and your Scientific Questions that they Answer

You will need to describe your ranking algorithms formally in your report in addition to implementing them. In addition you should describe and deliver the set of scientific questions that your algorithms answer based on your discussion with Dr. Burgess. Please see:

https://en.wikipedia.org/wiki/PageRank

For some guidance on describing your algorithms we suggest identifying as part of your description highlight which fields from the two approaches for indexing: (1) Solr index with SolrCell; and (2) Nutch/Tika and indexed into Solr you are leveraging in each of your relevancy algorithms, and also identifying which algorithms are more appropriate to answer each scientific question and why.

5. Report

Write a short 4 page report describing your observations. Initially begin with a description of your session with Dr. Burgess and explain how you came up with your scientific questions to ask of the data. In addition please answer how effective the link-based algorithm was compared to the content-based ranking algorithm in light of these scientific questions? What questions were more appropriate for the link based algorithm compared to the content one? Describe in detail and formally both of your ranking algorithms. You should describe the input, what your algorithms do to compute a rank, how to test them (and prove that they are working as expected). Do NOT simply provide advantages and disadvantages from a quick Google search. You are required to think critically about this portion of the report and sincerely provide your feedback.

Describe the indexing process – what was easier – Nutch/Tika + SolrIndexing; or SolrCell? Again a quick Google search will not suffice here. We are looking for your own words.
Please also note that the graders will be given great flexibility to add/subtract from various areas of the report based on your submitted work.

6. Submission Guidelines

This assignment is to be submitted **electronically, by 12pm PT** on the specified due date, via Gmail [csci572spring2015@gmail.com](mailto:csci572spring2015@gmail.com). Use the subject line: CSCI 572: Mattmann: Spring 2015: Solr Homework: <Your Lastname>: <Your Firstname>. So if your name was Lord Voldemort, you would submit an email to [csci572spring2015@gmail.com](mailto:csci572spring2015@gmail.com) with the subject “CSCI 572: Mattmann: Spring 2015: Solr Homework: Voldemort: Lord” (no quotes). **Please note only one submission per team.**

- All source code is expected to be commented, to compile, and to run. You should have a program developed in step #4b to annotate your Solr documents with your link based relevancy, and at a minimum should also have a program developed in step 6 that demonstrates your queries and the result sets they obtain.
- Include your program and/or script for generating MIME stats from a Nutch database. You may use Python and/or Java to generate this.
- Deliver your Solr configuration e.g., your schema.xml and solrconfig.xml files and any other configuration necessary to reproduce your results.
- Teams will be asked if they would like to contribute their indexed dataset to our NSF polar Amazon machine. This would include your Solr index. If you want to do this, please identify in your report that you would like to do this, and send a message to the professor, to the TAs, and to the graders.
- Also prepare a readme.txt containing any notes you’d like to submit.
- If you have used any external libraries in your two programs, you should include those jar files in your submission, and include in your readme.txt a detailed explanation of how to use these libraries when compiling and executing your program.
- Save your report as a PDF file (Lastname_Firstname_SOLR.pdf) and include it in your submission.
- Compress all of the above into a single zip archive and name it according to the following filename convention:
  
  `<lastname>_firstname_CSCI572_HW_SOLR.zip`

  Use only standard zip format. Do **not** use other formats such as zipx, rar, ace, etc.

**Important Note:**

- Make sure that you have attached the file when submitting. Failure to do so will be treated as non-submission.
- Successful submission will be indicated in the assignment’s submission history. We advise that you check to verify the timestamp, and download and double check your zip file for good measure.
• Again, please note, only **one submission per team**. Designate someone to submit.

6.1 **Late Assignment Policy**

• -10% if submitted within the first 24 hours
• -15% for each additional 24 hours or part thereof