Homework: Content extraction and search using Apache Tika – Employment Postings Dataset contributed via DARPA XDATA
Due: October 6, 2014 12pm PT

---

1. Overview

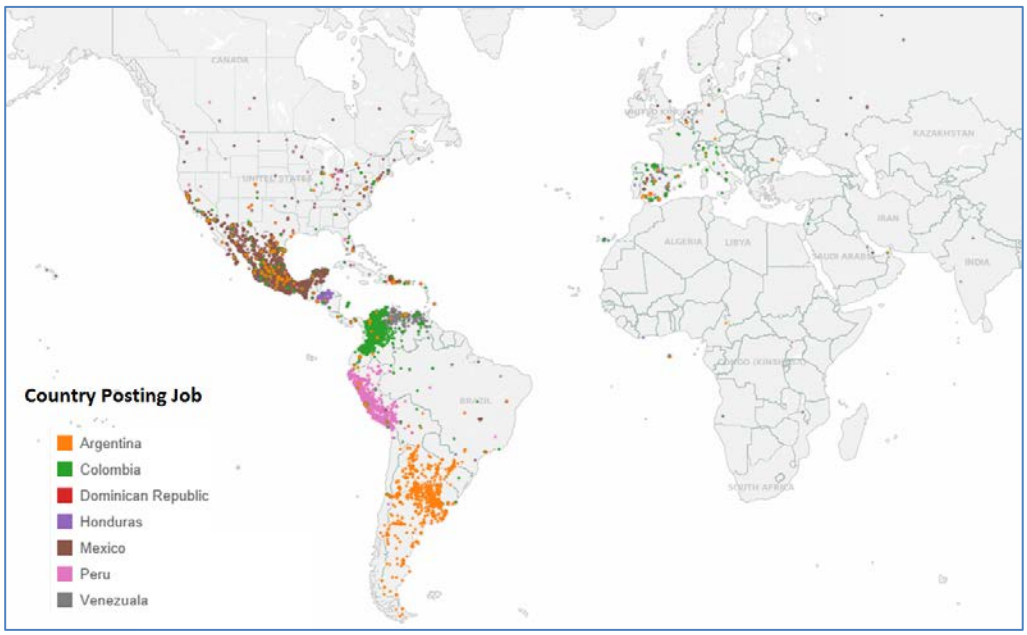

Figure 1: Map of Jobs (Colored by Country)

In this assignment you will participate in an ongoing effort to search job data from employment opportunities posted from http://www.computrabajo.com affiliate sites that primarily serve Mexico and South American countries. The dataset was contributed by the DARPA XDATA project an ongoing effort to develop open source big data analytics and visualization technologies for the government.

The postings have several properties that make them interesting and challenging from a search perspective. First, the postings are temporary and they may be taken down at any time, e.g., someone got the job; the employer decided to post a different job; the company no longer exists, etc. Second, the postings themselves tell a great deal about the economic and cultural information for a particular area. Third, the postings have data in Spanish due to their origins in Latin America, so understanding the information and searching it requires elements of *Machine Translation (MT)*. We won't get into the MT elements in this assignment; we will first stick to the searching, identifying, and deduplication activities. MT work will come in a later assignment when we look at methodologies and tools to convert the dataset from Spanish to English.

The dataset consists of ~119 million jobs and is currently ~40GB in size. The dataset is formatted as a set of Tab Separated Value (TSV) files.

2. Objective

You are going to create a local search engine of the Employment postings by cleansing, transforming, and developing an algorithm for ranking the job postings. TSV files are messy and difficult to understand, so your first task will be to transform the TSV files into Java Script Object Notation (JSON) files, and then from there into individual JSON posting files per job. You will initially operate on a subset of the data, but later in the assignment, you will run your search algorithm and process over the entire dataset.

To transform the TSV data, you will use Apache Tika (http://tika.apache.org/) and the ETLlib software package (http://github.com/chrismattmann/etllib/). More information about downloading Tika and getting started with ETLlib will be provided in the following section.

Tika will help you get the employment dataset into individual JSON records ("files") for searching. You will need to develop specific Tika support for TSV files as Tika currently parses those files as simple text, and does not directly support transformation of the data into JSON using its ContentHandler framework. After building this support into Tika, you will explore the use of the ETLlib package to perform a similar task. In addition, you will need to deduplicate the Employment job files – there is already much duplication and you can significantly reduce the size of the overall dataset by correctly performing deduplication.

Your final task on the assignment will be to search and analyze the dataset, as specified in the following section.

3. Tasks

1. Develop a Tika TSVParser that will
    a. Take in a TSVFile and create structured XHTML output, recognizing the column headers, and table row values.
2. Develop a JSONTableContentHandler that will
    a. Take the XHTML output from the TSVParser (or any upstream parser) and then output JSON files corresponding to the XHTML table rows (one file per row)
3. Develop and run a simple crawler that uses Tika across the initial reduced dataset of Employment jobs to produce the individual JSON job files
    a. How many job files were produced?
4. Develop and run a simple crawler that uses programs from the ETLLib software to output individual JSON job files from the initial reduced dataset of Employment jobs
    a. How many job files were produced?
5. Develop a process for deduplicating the Employment dataset
    a. Decide what job information can be used to deduce uniqueness
    b. Integrate deduplication into your programs from #3 and #4

    c. How many job files were produced when deduplication is enabled, and why?
6. **(EXTRA CREDIT)** Develop a program or a script to run Tika across the full dataset of Employment jobs to produce the individual JSON job files
    a. How many job files were produced (no deduplication)?
    b. Turn on deduplication – how many job files were produced?

4. Assignment Setup

4.1 Group Formation

You can work on this assignment in groups sized 2-4. Once you have decided upon your group please have one group member send one email to the following email addresses:

Gouthami Kondakindi kondakin@usc.edu
Preethi Ramesh pramesh@usc.edu;

Use subject: CS 572: Team Details

The email should consist of the names and email addresses of your group members.
This is due by September 18, 2014.

4.2 Downloading the Initial Dataset

The initial dataset is available at:
http://baron.pagemewhen.com/~chris/employment/

Each TSV file contains thousands of individual job records. You are responsible for:

1. Understanding the columns and format of the records. A list of the valid fields is provided below, part of your job is to map these columns to the actual data you are seeing in the file:
    a. postedDate
    b. location
    c. department
    d. title
    e. salary
    f. start
    g. duration
    h. jobtype
    i. applications
    j. company
    k. contactPerson
    l. phoneNumber:
    m. faxNumber:
    n. location

o. latitude
p. longitude
q. firstSeenDate
r. url
s. lastSeenDate

2. Converting the TSV file into a set of individual employment job JSON files
3. Analyzing the dataset
4. Answering the questions from #3

4.3 Downloading Apache Tika

The quickest and best way to get Apache Tika up and running on your machine is to grab the `tika-app.jar` from: http://tika.apache.org/download.html. You should obtain a jar file called `tika-app-1.6.jar`. This jar contains all of the necessary dependencies to get up and running with Tika by calling it your Java program.

Documentation is available on the Apache Tika webpage at http://tika.apache.org/. API documentation can be found at http://tika.apache.org/1.6/api/

You can also get more information about Tika by checking out the book written by Professor Mattmann called "Tika in Action", available from: http://manning.com/mattmann/.

4.4 Downloading ETLlib

ETLlib (https://github.com/chrismattmann/etllib) is a Python based toolkit for extract, transform and load operations around file based data. There are command line tools and APIs for manipulating data in TSV and in JSON.

You can install and download ETLLib per the instructions, here:

https://github.com/chrismattmann/etllib/blob/master/README.md

5. Report

Write a short 4 page report describing your observations, i.e. what you noticed about the dataset as you answered the questions in Part #3. Why do you think there were duplicates? Were they easy to detect? Describe your algorithm for deduplication. How did you arrive at it? What worked about it? What didn't? Describe your simple crawlers. What could they do better?

Also include your thoughts about Apache Tika and ETLlib – what was easy about using them? What wasn't?

6. Submission Guidelines

This assignment is to be submitted ***electronically, by 12pm PT*** on the specified due date, via Gmail csci572fall2014@gmail.com. Use the subject line: CSCI 572: Mattmann: Fall 2014: Tika Homework: <Your Lastname>: <Your Firstname>. So if your name was Lord

Voldemort, you would submit an email to csci572fall2014@gmail.com with the subject "CSCI 572: Mattmann: Fall 2014: Tika Homework: Voldemort: Lord" (no quotes).

- All source code is expected to be commented, to compile, and to run. You should have (at least) one Java source file and should also include other java source files that you added, if any. Do **not** submit `*.class` files. We will compile your program from submitted source.
- Include your `colheaders.txt` file containing the the TSV column headers used in ETLlib.
- Also prepare a `readme.txt` containing any notes you'd like to submit.
- Do **not** include `tika-app-1.6.jar` and the TSV files in your submission. We already have these.
- However, if you have used any external libraries other than Tika, 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_TIKA.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_TIKA.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 the 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, download and double check your zip file for good measure.

6.1 Late Assignment Policy

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