

CPTS 415 Big Data Course Project

This document lists some suggested projects. You are also encouraged to propose your own project. The project must tackle a data science problem using both a sequential algorithm and a parallel / distributed algorithm / platform (Apache Spark, Hadoop, GraphX, Giraph). For your own project, you may come up with any research topic/problem that you think is interesting by using the list of available datasets below or other dataset you collected. You should talk with the instructor first about your proposed project, including the dataset you selected.

Project 1: YouTube Analyzer

Related Industry: Social Media

Data: <http://netsg.cs.sfu.ca/youtubedata/>

Suggested Problem: Implement a Youtube data analyzer supported by MapReduce, SQL and/or graph algorithms. The analyzer provides basic data analytics functions to Youtube media datasets. The analyzer provides following functions for users:

- A. Network aggregation: efficiently report the following statistics of Youtube video network: -
 - Degree distribution (including in-degree and out-degree); average degree, maximum and minimum degree
 - Categorized statistics: frequency of videos partitioned by a search condition: categorization, size of videos, view count, etc.
- B. Search.
 - top k queries: find top k categories in which the most number of videos are uploaded; top k rated videos; top k most popular videos;
 - Range queries: find all videos in categories X with duration within a range [t1, t2]; find all videos with size in range [x,y].
 - User identification in recommendation patterns: find all occurrence of a specified subgraph pattern connecting users and videos with specified search condition.

*develop effective optimization techniques to speed up the algorithm you used, including indexing, compression, or summarization.

C. Influence analysis.

- Use PageRank algorithms over the Youtube network to compute the scores efficiently. Intuitively, a video with high PageRank score means that the video is related to many videos in the graph, thus has a high influence. Effectively find top k most influence videos in Youtube network. Check the properties of these videos (# of views, # edges, category...). What can we find out? Present your findings.

Project 2: Airline Search Engine

Related Industry: Aviation

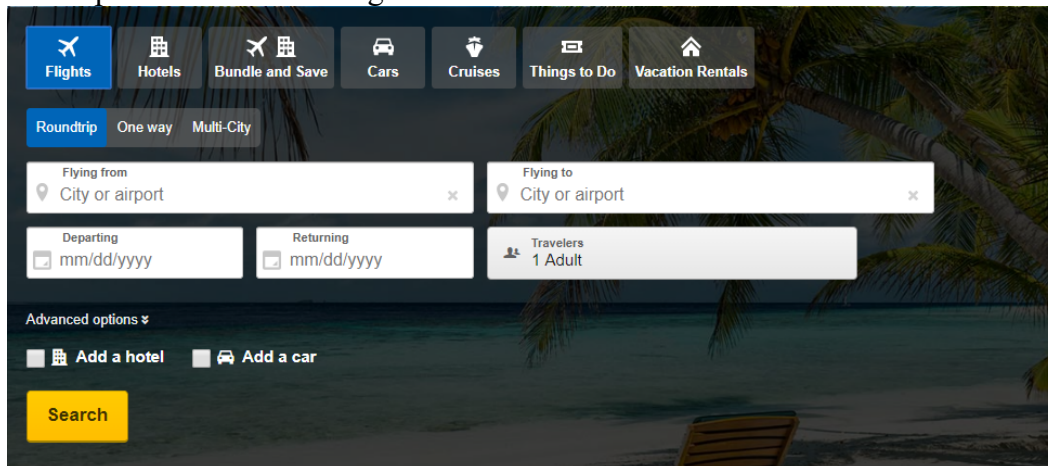
Data: <http://openflights.org/data.html>

Public available dataset which contains the flight details of various airlines like: Airport ID, Name of the airport, Main city served by airport, Country or territory where airport is located, Code of Airport, Decimal Degrees, Hours offset from UTC, Timezone, etc.

Suggested Problem: Implement an airline data search engine supported by efficient MapReduce, SQL/SPARQL and/or graph algorithms.

The tool is able to help users to find out facts/trips with requested information/constraints:

- Airport and airline search:
 - o Find list of airports operating in the Country X
 - o Find the list of Airlines having X stops
 - o List of airlines operating with code share
 - o Find the list of active airlines in the United States - Airline aggregation:
 - o Which country (or) territory has the highest number of Airports.
 - o Top K cities with most incoming/outgoing airlines
- Trip Recommendation:
 - o Define a trip as a sequence of connected route. Find a trip that connects two cities X and Y (reachability).
 - o Find a trip that connects X and Y with less than Z stops (constrained reachability)
 - o Find all the cities reachable within d hops of a city (bounded reachability).
 - o Fast Transitive closure/connected component implemented in parallel/distributed algorithms



The image shows a flight search interface with a dark background and a tropical beach scene. At the top, there are several tabs: Flights (selected), Hotels, Bundle and Save, Cars, Cruises, Things to Do, and Vacation Rentals. Below these are trip type options: Roundtrip (selected), One way, and Multi-City. The main search area includes fields for 'Flying from' (City or airport), 'Flying to' (City or airport), 'Departing' (mm/dd/yyyy), 'Returning' (mm/dd/yyyy), and 'Travelers' (1 Adult). There are also checkboxes for 'Add a hotel' and 'Add a car' under 'Advanced options'. A yellow 'Search' button is at the bottom left.

Project 3: Amazon co-purchasing analysis

Related Industry: online commercial/Business

Data: <http://snap.stanford.edu/data/amazon-meta.html>

The data was collected by crawling Amazon website and contains product metadata and review information about 548,552 different products (Books, music CDs, DVDs and VHS video tapes).

Suggested Problem: Implement a co-purchasing data analytics engine. The analyzer has the following functions.

1. Answer complex query. We define a SQL-like query Q of the form SELECT* FROM U WHERE Condition. The CONDITION is of the following forms:
 - Searchable attributes: value constraints over well-defined attributes in node/edge schema
 - Non-searchable attributes: attributes that cannot be queried directly over existing attributes: the number of reviews of a product, the number of customers copurchasing same product of a user.
 - Queries with enriched operators: >, >=, =, <, <=; e.g., Select movie with average rating >=4.5

Given a query Q and Amazon dataset, and a number k, find k entities that satisfy Q with minimized evaluation cost.

2. Find potential customers that satisfies co-purchasing pattern. Divide the co-purchasing data into two data set, one we call “training” dataset, and the other “testing” dataset. Verify several frequent co-purchasing patterns in the training dataset. Report the frequency in the testing dataset. For those frequent patterns in both dataset, return the customers captured by the patterns. What seems to be the most significant co-purchasing pattern? (e.g., collaborative filtering)

Project 4: Geospatial Hot Spot Analytics

Related industry: location intelligence companies such as Uber, Lyft and Google Maps

Data:

The data contains the taxi trip records of New York City in January 2009.

Suggested problem:

This task will focus on applying spatial statistics to spatio-temporal big data in order to identify statistically significant spatial hot spots using Apache Spark. The topic of this task is from ACM SIGSPATIAL GISCUP 2016.

The Problem Definition page is here: <http://sigspatial2016.sigspatial.org/giscup2016/problem>

The Submit Format page is here: <http://sigspatial2016.sigspatial.org/giscup2016/submit>

Note that: You may clip the source data to an envelope (latitude 40.5N – 40.9N, longitude 73.7W – 74.25W) encompassing the New York City in order to remove some of the noisy error data.

Special requirement (different from GIS CUP):

As stated in the Problem Definition page, in this task, you are asked to implement a Spark program to calculate the Getis-Ord statistic of NYC Taxi Trip datasets. We call it "**Hot cell analysis**"

To reduce the computation power need, we made the following changes:

1. The input will be a monthly taxi trip dataset, "yellow_tripdata_2009-01_point.csv"
2. Each cell unit size is 0.01 * 0.01 in terms of latitude and longitude degrees.
3. You should use 1 day as the Time Step size. The first day of a month is step 1. Every month has 31 days.
4. You only need to consider Pick-up Location.

Available dataset (also see “resource” on the course homepage)

<https://github.com/awesomedata/awesome-public-datasets>

<https://snap.stanford.edu/data/>

Project milestones

1. Proposal: Select a project and understand the dataset, or come up with your own project over the dataset list. Formulate your problem, and review related work.
2. Data preparation: Prepare data collection and formatting. Description of data collection and the tools you use. Usually you will write a parser to extract the information you need to the data structure/platform you will be using.
3. Algorithm: Description of any mathematical background and data statistics from your dataset. Formal description of the algorithms you developed
4. Final report and presentation: Experimental study/Demo and justify the result with baseline methods.

Project report

The final project report should be a 5-10 page paper, describing the introduction, related work, approach, results and conclusion. We will not accept reports longer than 10 pages. At the end of the report, you should also highlight the contributions of individual team members to the project (in the format outlined below). The project report should contain at least some amount of algorithm analysis, and some experimentation on real or synthetic data.

I will use the following guidelines when grading your final project write-ups. Keep in mind however, that if there is a good reason why your project doesn't match the rubric below, we will take that into consideration when grading your report. For example, we recognize that purely theoretical or pure data analysis projects may not fit the rubric below perfectly, and that depending on your project you may want swap the ordering of certain sections. But hopefully all projects can be roughly mapped to the criteria below:

- Introduction/Motivation/Problem Definition (15%): What is it that you are trying to solve/achieve and why does it matter.
- Related Work (10%): How does your project relate to previous work. Please give a short summary on each paper you cite and include how it is relevant.
- Model/Algorithm/Method (30%): This is where you give a detailed description of your primary contribution. It is especially important that this part be clear and well written so that we can fully understand what you did.
- Results and findings (35%): How do you evaluate your solution to whatever empirical, algorithmic or theoretical question you have addressed and what do these evaluation methods tell you about your solution. It is not so important how well your method performs but rather how interesting and clever your experiments and analysis are.
- We are interested in seeing a clear and conclusive set of experiments which successfully evaluate the problem you set out to solve. Make sure to interpret the results and talk about what can we conclude and learn from your evaluations. Even if you have a

theoretical project you should have something here to demonstrate the validity or value of your project (for example, proofs or runtime analysis).

- Style and writing (10%): Overall writing, grammar, organization and neatness.

Unlike the project proposal and milestone, we plan to assign individual scores to team members for the final project report. We observed that there is a skewed distribution of work in some of the teams and would like to take that into account when we are grading. Your score for the final report will now be a function of two aspects:

- The criteria outlined above for your final report
- Your contribution to the project relative to that of your team members.

In order to be able to assign such individual scores, we want you to write down a brief summary of the individual contributions of each of the team members.

Project competition

We plan to have a project competition among different teams. After the final presentation, we will ask all students in this class to rate your project according to the algorithm contribution and practical applications. If you are ranked top 25% among all groups, all of your group members will receive 3 extra credits which can be directly applied to your final grade.

Samples of Previous Big Data Course Projects:

Team 1: "Airline search engine"

Team 2: "Analysis of The Social network climate of Stack Overflow"

Team 3: "Fraudulent reviewers/reviews detection in YELP"

Team 4: "Resource-bounded query evaluation"

Team 5: "Knowledge base fact checker"

Team 6: "Knowledge base search engine"

Team 7: "Youtube analyzer"