CPTS 415 Big Data
Course Project

This document lists three suggested projects. You are also encouraged to propose your own project. 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
Publicly 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:
    1. Find list of Airports operating in the Country X
    2. Find the list of Airlines having X stops
    3. List of Airlines operating with code share
    4. Find the list of Active Airlines in the United States
- Airline aggregation:
    5. Which country (or) territory has the highest number of Airports
    6. The top k cities with most incoming/outgoing airlines
- Trip recommendation:
    6. Define a trip as a sequence of connected route. Find a trip that connects two cities X and Y (reachability).
    7. Find a trip that connects X and Y with less than Z stops (constrained reachability).
    8. Find all the cities reachable within d hops of a city (bounded reachability).
    9. *Fast Transitive closure/connected component implemented in parallel/distributed algorithms.

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 co-purchasing 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?

Project 4: Collaboration Analysis
Related Industry: Publisher/Scholar/Academic
Data: Microsoft Academic Graph (https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/)
DBLP (http://dblp.uni-trier.de/xml/)

Suggested Problem: How to discover interesting (potential) collaboration among scientists and researchers? How to track the "strong" collaboration among authors and find out the reason why some collaboration success and some is not trending? An academic collaboration analyzer gives possible answers.

1. Find dense communities of interests. Given a set of labels denoting research domains/topics of papers etc. from end users, the goal is to discover dense subgraph/community of the citation network that closely connects all the entities satisfying the labels, and summarize it as a collaboration pattern. A collaboration pattern connects a set of authors, papers and the venues/conferences/journals the papers are published.

2. Tracking the dynamics in citation networks. Given a collaboration pattern, how does the support information changes over time? How to discover special time point/outlier/anomalies/events over the evolving citation network for this pattern?

3. Association and correlation analysis. Given a set of keywords describing research area/topics, define and discover association rules that specify the correlation/regularities among the entities (authors, papers, conferences, journals, universities) that are similar to the keyword description.

4. Link prediction. The rules mined in 3 suggests possible facts and collaborations in near future. Evaluate the "power of prediction" of the rules you discovered in the experimental study, and report the interesting findings (accuracy, confidence, support).

Project 5: Panama Offshore Leaks
Related Industry: Financial/Economics/Politics
Data:ICIJ Offshore Leak. http://www.thereportertimes.com/panama-papers-icij-offshore-leaks-database-documents/23489/

The ICIJ Offshore Leaks Database is licensed under the Open Database License and its contents under Creative Commons Attribution-ShareAlike license. Always cite the International Consortium of Investigative Journalists when using this data. This database is powered by Neo4j, a graph database that structures data in nodes (the icons you see in the visualization) and relationships (the links between nodes).

Suggested Problem: 1. A representation of Panama Papers into a network of person, companies, relationships and timestamps. 2. Mining frequent graph patterns that suggests interesting activities, potential anomaly events and outliers over snapshots of the Panama paper network. 3*. Develop algorithms that track the information of these patterns (e.g., close/reopen of companies, tax heaven, abnormal shutdown of companies, social network of users and supervisions).

Project 6*: Knowledge base fact checker
Related Industry: Search engines/General database/knowledge base applications
Data: DBPedia: http://wiki.dbpedia.org/ Richly labeled network containing extracted data from Wikipedia (based on infoboxes). Labeled network of multiple types of nodes and edges About 2.6 million concepts described by 247 million triples, including abstracts in 14 different languages.
YAGO(http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/downloads/)

Suggested Problem: Fact checking is a critical task in knowledge base quality management. Given a knowledge base (KB) as a knowledge graph, and a set of edges indicating facts, the task is to determine whether these facts are (likely) to be true or not.

Functional dependencies have been used to describe hard constraints among attribute values of the tuples in relational databases. The FDs are used to detect violations of dependencies that often indicate inconsistencies/quality issues. This course project requires you to make use of the idea of FDs and its variants (CFDs and GFDs) to come up with rules that decides if a fact holds, and develop an algorithm that discover these rules over an open KB dataset.

References: http://homepages.inf.ed.ac.uk/wenfei/papers/vldb15-GPAR.pdf

Project 7*: Knowledge base search engine
Related Industry: Search engines/General database/knowledge base applications
Data: DBPedia/YAGO

Suggested Problem: Keyword search has been redefined by modern search engines: users wants to search for “things”, not “strings”. An efficient keyword search over a knowledge graph provides answers in terms of a small subgraph, rather than a set of keyword/text. The outcome of such search engines directly applies to Q&A systems, AI systems and smart environment.

Given a (fraction) of knowledge graphs, develop and implement a KB search engine that takes as input a set of (possibly ambiguous) keywords, and outputs a small subgraph of interests that contains matches of the keywords. In this project, you need to consider the following questions: (1) how to define a similarity/clustering measure of the relevant entities? (2) how to define the relevance of the answers? And (3) creatively apply related research on “keyword search over graph” to perform the search.

Reference: http://link.springer.com/chapter/10.1007%2F978-1-4419-6045-0_8#page-1

Project 8*: Making search bounded over Big Data
Related Industry: General
Data: General

Suggested Problem: In this class we will introduce a class of strategies to make query processing "bounded". That is, given a certain resource bound over e.g., running time, # of entities you are allowed to fetch, etc, develop an algorithm that finds the best answers for a given query within the resource budget.

This is an open project. In this project, you identify a query class, a dataset, and develop a resource-bounded search algorithm that makes use of big data processing/querying strategies (indexing, compression, query evaluation using views, caching, sketching, sampling, sparsifying, filtering, selection…) You should justify the performance of your proposed algorithm for the query class. The method you propose should be general enough that apply to every query from the query class.

Reference: http://homepages.inf.ed.ac.uk/wenfei/papers/sigmod14.pdf

Available dataset (also see "resource" on the course homepage)

Project milestones

1. 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. 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. Description of any mathematical background and data statistics from your dataset
4. Formal description of the algorithms you developed
5. 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 mathematical 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 do 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.