CSE 515 Multimedia and Web Databases

Phase #3

(Due December 5th 2014, midnight)

Description: In this project, you will experiment with

- indexing,
- graph analysis, and
- classification.

This project will build on the deliverables of the previous phase. Consider the data that was provided to you in the previous phase. Remember from the previous phases of the project that epidemic word files consist of pairs

$$w_i = \langle idx_i, \vec{win}_i \rangle,$$

where idx_i is a file-state-time triple of the form

$$idx_i = \langle f, s, t \rangle$$

.

- **Task 1: Multi-dimensional index structures and nearest neighbor search task I:**
  - Implement a Locality Sensitive Hashing (LSH) tool, which takes as input (a) the number of layers, L, (b) the number of hashes per layer, k, and (c) an epidemic word file as input and creates an in-memory index structure containing the indexes of the given set of vectors. See

    ”Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions” (by Alexandr Andoni and Piotr Indyk). Communications of the ACM, vol. 51, no. 1, 2008, pp. 117-122.

    The program also outputs the size of the index structure in bytes.
  - Implement a similarity-based epidemic simulation search program using this index structure: for a given simulation and t, the program outputs the t most similar epidemic simulations. The program also outputs (a) the number of unique vectors considered, (b) the overall number of vectors considered, and (c) the number of bytes of data from the index accessed to process the query.
- **Task 2: Multi-dimensional index structures and nearest neighbor search task II:**
  - Implement a VA-file index tool and associated nearest neighbor search operations. The data structures and relevant algorithms are described in the following two papers:

    Stephen Blott and Roger Weber, “A Simple Vector-Approximation File for Similarity Search in High-Dimensional Vector Spaces” 1997. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.9708

    Roger Weber, Hans-Jörg Schek, and Stephen Blott. 1998. “A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces”. In Proceedings of the 24rd International Conf erence on Very Large Data Bases (VLDB ’98), pp. 194-205. 1998. http://dl.acm.org/citation.cfm?id=671192

Given (a) a parameter b denoting the number of bits per dimensions used for compressing the vector data and (b) an epidemic word file as input, the program and creates an in-memory index structure containing the indexes of the given set of vectors. The program also outputs the size of the index structure in bytes.

- Implement a similarity-based epidemic simulation search program using this index structure: for a given simulation and t, the program outputs the t most similar epidemic simulations. The program also outputs (a) the number of bytes of data from the index accessed to process the query and (b) the number of compressed vectors needed to expand to answer the query.

- **Task 3: Epidemic simulation graph analysis task:**
  - (a) Implement a program which creates a weighted epidemic simulation similarity graph, where there is an edge between two simulations if their similarity is beyond a given threshold τ.
  - (b) Implement a program which, given a simulation similarity graph, identifies and visualizes K most dominant nodes using Page Rank (PR) for a user supplied K.
  - (c) Implement a program which, given a simulation similarity graph and two simulation files, identifies and visualizes K most relevant simulation files using personalized PageRank (PPR) for a user supplied K.

  See
  - "J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In KDD, pages 653658, 2004"

  for a personalized PageRank formulation based on Random Walks with re-start.

- **Task 4: Epidemic simulation classification:** Implement
  - (a) a k-nearest neighbor based classification program and
  - (b) a decision tree based classification program

  each of which takes a set of labeled epidemic simulations and associates labels to the rest of the epidemic simulations in the database.

  See Chapters 9.1 and 9.2.

  Deliverables:
  - Your code (properly commented) and a README file.
  - Your outputs for the provided sample inputs.
  - A short report describing your work and the results.

  Please place your code in a directory titled "Code", the outputs to a directory called "Outputs", and your report in a directory called "Report"; zip or tar all off them together and submit it through the digital dropbox.