CSE 515 Multimedia and Web Databases

Phase #3

Due (Tuesday) November 28 2023, midnight

Description: In this project, you will experiment with

- clustering,
- indexing,
- classification / relevance feedback

NOTES:

- We will continue using the Caltec101 data set
- The tasks in this phase involve the feature models, similarity/distance functions, and latent space extraction algorithms developed in the previous phase.
- Aside from very low level math libraries, no other libraries can be used.

PROJECT TASKS:

- **Task 0a:** Implement a program which computes and prints the "inherent dimensionality" associated with the even numbered Caltec101 images.
- **Task 0b:** Implement a program which computes and prints the "inherent dimensionality" associated with each unique label of the even numbered Caltec101 images.
- **Task 1:** Implement a program which,
  - for each unique label l, computes the corresponding k latent semantics (of your choice) associated with the even numbered Caltec101 images, and
  - for the odd numbered images, predicts the most likely labels using distances/similarities computed under the label-specific latent semantics.

  The system should also output per-label *precision*, *recall*, and *F1-score* values as well as output an overall *accuracy* value.
- **Task 2:** Implement a program which,
  - for each unique label l, computes the corresponding c most significant clusters associated with the even numbered Caltec101 images (using DBScan algorithm); the resulting clusters should be visualized both
    - as differently colored point clouds in a 2-dimensional MDS space, and
    - as groups of image thumbnails.

    and

  - for the odd numbered images, predicts the most likely labels using the c label-specific clusters.

  The system should also output per-label *precision*, *recall*, and *F1-score* values as well as output an overall *accuracy* value.

- **Task 3:** Implement a program which,
  - given even-numbered Caltec101 images,
    * creates an m-NN classifer (for a user specified m),
    * creates a decision-tree classifier,
    * creates a PPR based clasifier.

    For this task, you can use feature space of your choice.
  - for the odd numbered images, predicts the most likely labels using the user selected classifier.

  The system should also output per-label *precision*, *recall*, and *F1-score* values as well as output an overall *accuracy* value.

- **Task 4:**
  - **4a:** Implement a Locality Sensitive Hashing (LSH) tool (for Euclidean distance) which takes as input (a) the number of layers, L, (b) the number of hashes per layer, h, and (c) a set of vectors as input and creates an in-memory index structure containing 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.
  - **4b:** Implement a similar image search algorithm using this index structure storing the even numbered Caltec101 images and a visual model of your choice (the combined visual model must have at least 256 dimensions): for a given query image and integer t,
    * visulizes the t most similar images,
    * outputs the numbers of unique and overall number of images considered during the process.

- **Task 5:** Let us consider the tag set "Very Relevant (R+)", "Relevant (R)", "Irrelevant (I)", and "Very Irrelevant (I-)". Implement
  - an SVM based relevance feedback system,
  - a probabilistic relevance feedback system – see

    Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. Journal of the American Society for Information Science. 41, pp. 288-297, 1990.

  which enable the user to tag some of the results returned by 4b as and then return a new set of ranked results, relying on the feedback system selected by the user, either by revising the query or by re-ordering the existing results.

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.