CSE225 Data Structures, 2017(FALL)

PROJECT #2 (Deadline:18.11.2017, Saturday, 24:00; Demos:20-21.11.2017)

BST vs AVL

This project is a programming assignment in C which aims to build two algorithms (BST and AVL) that will store/search/update/delete information related to textual materials and make performance experiments in order to evaluate your system.

Steps:

1. Read the Input1.txt.
2. Make necessary stop word removal and stemming operations.
3. Store your data into **BST**.
4. Delete the keys in the first sentence of Input1.txt from your BST and re-store your tree.
5. Record your memory usage, execution time in the 3rd and 4th steps.
6. Read the Input1.txt.
7. Make necessary stop word removal and stemming operations.
8. Store your data into **AVL tree**.
9. Delete the keys in the first sentence of Input1.txt from your AVL tree and re-store your tree.
10. Record your memory usage, execution time in the 8th and 9th steps.
11. Compare the results you get in 5th and 10th steps.
12. Do the steps 1st-11th for Input2.txt.

Input1.txt

Text mining studies have gained importance in recent years because of the increasing number of electronic documents like news, social networks, research papers and digital libraries. There is no doubt that this enormous data continues to increase day by day with the contribution of lots of people. Automatically processing, organizing and handling this textual data are a central problem. The key aim of text mining is to allow users to get information from textual materials. Text mining mainly deals with several important applications like information retrieval (IR), classification (i.e., supervised, unsupervised and semi supervised classification), document filtering, summarization, sentiment or opinion classification. Natural Language Processing (NLP), Machine Learning (ML) and Data Mining methods work together to detect patterns from the different types of the documents and classify them in an automatic manner.

Input2.txt

An expected output of accurate and very efficient text classification algorithms is to label unlabeled textual materials based on specified classes that comprise identical textual materials. In order to accomplish this goal, there are various classification methods which are

based on similarity measures. These similarity measures compare pairs of documents and compute their similarities. Vector space demonstration of texts results in sparsity and high dimensionality. This is a very big difficulty especially when there are numerous class labels but inadequate training data. Hence it is critical that a successful and accurate text classifier should scale well with the large number of classes and features under the circumstances of restricted training data. However, rather preferably, terms in documents convey semantic information. Accordingly, a perfect text classifier should have the capability of using this semantic information.

CODE SUBMISSION:
You should use the following email address in order to submit your code:
cse225.marmara.2017 at gmail dot com
naming standart:

name_surname.c

Your any submission after the project submission due date, will not taken into consideration.

You are required to exhibit an ***individual effort*** on this project. Any potential violation of this rule will lead everyone involved to **failing from the course** and necessary disciplinary actions will be taken.
Good luck!!!