Search engine project

1- Introduction

With the explosion of the number of media content (text, image, video), there is a necessity to have algorithms that can access data in a fast way. This can be done with search engines.

Web search engines such as Google is the most important application of information retrieval. However, search engines are also commonly used in companies for many purposes (document search for example laws, product search, ...).

2- Description of the project

The main focus will be on document search (text). The goal is to return a ranked list of the most relevant documents for a query. The process is composed of 2 main steps:

- Indexing
- Searching

The indexing part consists of 3 steps:

- Text acquisition (ex crawling on the web to collect the web pages)
- Text transformation (Processing of text to a list of indices)
- Index creation (data structure used for fast search)

In this project, the text acquisition part will be skipped and a pre-defined corpus of text will be used.

The searching part also consists of three 3 steps:

- Ranking (main part : returns the list of relevant documents)
- Evaluation (Relevance of the documents returned thanks to user logs)
- User Interaction (Take into account user choices, personalization, ...)

The evaluation and user interaction part will not be treated in this project as they require log informations which are not available.

The 1st part of the project will consist of a basic implementation of a search engine. The 2nd part will consist on improvements of the search engine (to improve performances on indexation, query processing, ...) or an extension to image search.

1st part of the project

The groups will implement a basic search engine. It will have the following functionalities:

- Functions to clean and preprocess text
- Function to create the inverted index
- Functions to save and load the index
- Different functions to make the search process

The search engine will be based on the vector space model (cosine similarity) using term frequency - inverse document frequency (tf-idf) features.

The notation will be based on the following criteria:

- Correctness of the algorithm
- Clean and well-tested code
- Analysis of the results

The analysis of the results includes an evaluation of the performances of the model:

- The relevance of the documents retrieved
- The speed of the index creation and search process (depending on different factors)
- The size of the index

2nd part of the project

The 2nd part of the project can consist of improvements on the search engine or an extension to image search.

1- Improvements of the search engine

The major improvements that can be done are on 3 points:

- Making a scalable index creation algorithm
- Compression of the index size
- Faster search

For the 1st improvement, distributed libraries such as Spark could be used with algorithms based on sorting and merging.

The major method for the index compression is gap compression (save the difference between doc ids instead of doc ids) and variable length codes.

Finally, faster searches can be obtained thanks to pruning. As we are only interested in the k best results, some documents can be eliminated during the search process because they are not promising enough.

The evaluation of this part will be based on the new features added to the search engine and the analysis of the new performances.

2- Image search

The first part of image search is to extract local descriptors on the image. This is often done using the SIFT algorithm. It makes it possible to extract in a robust manner interest points. His descriptors have nice properties such that invariance to rotation and scale.

The second part is to quantify the descriptors into visual words. It can be done using a clustering algorithm. It’s an algorithm that associates similar descriptors in a same group. The most used one is K-means. Each descriptor will be associated to a cluster (visual word). As for text, each descriptors will be counted and indexed.

For the search, the descriptors of the query image must be calculated and quantified. The quantification consists in searching the nearest visual word of the descriptor. The rest of the search process is exactly the same as for text.

For this part, a dataset and the associated image descriptors will be given. The students should evaluate the performance of their model thanks to the groundtruth.

Evaluation of the project

The project will be evaluated on:

- The source code of the project (must be documented and tested)
- An analysis of the search engine performances (text or/and image) and of the project in general
- The presentation of your project which includes a demonstration of the search engine and an analysis of your project.