Search Engine Architecture

The goal of this class is to implement a highly-scalable and highly-available search engine by leveraging off-the-shelf tools. We will use elements of information retrieval, NLP, machine learning, and large-scale cluster computing with a focus on practical implementation. Prerequisites are an understanding of basic data structures and algorithms, and working knowledge of Python.

Textbook: Introduction to Information Retrieval, Christopher D. Manning.

- Introduction
    - Goals of search engines and their architectures
    - Component and dataflow overview
- Basic modeling
    - Jaccard coefficient, vector space, cosine similarity, TF-IDF weighting, evaluation
    - sklearn, NLTK
- Processing
    - Crawling, filtering noise, tokenization, stemming, phonetic algorithms
    - Scalability: Hadoop, HDFS, HBase
- Indexing
    - Inverted indexes, locality-sensitive hashing: Redis
    - Distributed hash tables: Chord
- Searching
    - Reverse proxies: HAProxy
- Collaborative filtering
    - Nonnegative matrix factorization, nearest-neighbor methods
- Link Analysis
    - PageRank
- Text categorization
    - Discriminative methods
- Text clustering
    - Latent semantic indexing, dimensionality reduction, latent Dirichlet allocation