Web Search
Course presentation
João Magalhães
A bit of history – 20th Century

• 1945 Vannevar Bush’s Memex (Memory Extender)
• 1965 Ted Nelson’s Hypermedia
• 1975 Keith v. Rijsbergen’s “Information Retrieval” book
• 1989 Tim Berners-Lee’s first HTTP and HTML implementations
• 1992 Lynx Web Browser
• 1993 NCSA Mosaic Browser, first TREC
• 1994 Netscape, Yahoo!, WebCrawler, ...
• 1995 Altavista search engine, Apache, PHP
• 1996 Wayback machine
• 1997 PageRank
A bit of history – 21st Century

• 2001 Wikipedia
• 2003 MySpace, Hi5, Skype
• 2004 Facebook, Flickr
• 2005 YouTube, Reddit, Mechanical Turk
• 2007 Twitter
• 2010 Instagram, Kaggle
• 2011 SnapChat
• 2012 Vine
• ...
Relevance vs similarity

What is the best [search space + dissimilarity function] to compute the relevance of documents for a given user information need?
What makes a good search application?

• **Efficiency**: application replies to user queries without noticeable delays.
  - 1 sec is the “limit for users feeling that they are freely navigating the command space without having to unduly wait for the computer”

• **Effectiveness**: application replies to user queries with relevant answers.
  - This depends on the interpretation of the user query and the stored information.
The tasks of a search application

• **Collect** data for storage
  • Crawler

• Analyse collected data and compute the **relevant information**
  • Information analysis

• Store data in an **efficient** manner
  • Indexing

• Process **user** information needs
  • Querying

• Find the documents that best **match** the user information need
  • Ranking
Crawler

• Discovers data for storage and indexing

• Applied when data has to be “discovered”
  • i.e., only a sample of the full data can be collected.

• If the sample is unbiased, it is a faithful representation of the real scenario.

• Implements a strategy to collect relevant data
  • e.g. on the Web the crawler needs to decide which links are more fruitful to follow and if a page should be indexed or not (it can be spam or phishing)
Web crawling

Begin with known “seed” URLs
Fetch and parse them
  Extract URLs they point to
  Place the extracted URLs on a queue
Fetch “robots.txt”
Fetch each URL on the queue and repeat
Information analysis

• This stage deals with the extraction of the information to be made searchable

• Extract meaningful words, pairs of words or n-grams

• Extract images and their main characteristics

• Link visual characteristics and text data
Indexing

• This stage creates an index to quickly locate relevant documents

• An index is an aggregation of several data structures (e.g. several B-trees)

• Index compression is used to reduce the amount of space and the time needed to compute similarities

• The distribution of the index pages across a cluster improves the search engine responsiveness
Querying

• Conversion of the user query into the internal search space
  • Parsing

• Usage history
  • Cookies, profiles, etc.

• User intention
  • What type of task is the user doing?
Ranking

• Once the user query is converted into the internal search space...
  • The ranking function sorts the information according to its relevance to the user query

• Ranking functions should model the human notion of relevance
  • We don’t really know the mathematical form of the human notion of similarity... it is highly subjective and dynamic. 😊

Since all models are wrong, one can only hope for useful approximations.
Putting all together...
References

• Slides and articles provided during classes.

• Books:


Course grading

• 40% theoretical part (2 tests or 1 exam)

• Labs (groups of 3 students):
  • 20% Checkpoints
  • 40% Project

• Project submission and exam dates:
  • First checkpoint 4 Oct
  • Second checkpoint 6 Nov
  • Project 18 December
  • First test 23 October
  • Second test 11 December
  • Exam To be defined
Checkpoints: StackOverflow answer search

• Task:
  • Implement a search engine to search for StackOverflow CrossValidated Answers.
  • Understand the roles of each component of a search engine in the performance of the search results.

• Checkpoints:
  • First: Baseline provided in the first class + results visualizations
  • Second: Comparison of three search methods
Project: Information stream summaries

- Task: for a given query, provide a temporal summary of information
Project grading

• Scoring
  • Imp. originality: 25%
  • Code “cleanness”: 25%
  • Critical discussion: 25%
  • Report: 25%

• ACM template format (word or latex) and 8 pages maximum (no-cover, references and annexes don’t count to the page limit).

• Report organization:
  • Introduction
  • Algorithms from classes used in the project
  • Implementation:
    • What are your ideas? What makes your project unique?
  • Evaluation
    • Dataset description
    • Baselines
    • Results
  • Critical discussion
  • References
## Classes plan (check also the deadlines)

<table>
<thead>
<tr>
<th>Date</th>
<th>#</th>
<th>Lectures</th>
<th>#</th>
<th>In-class labs</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-Sep-17</td>
<td>1</td>
<td>Introduction</td>
<td>1</td>
<td>Setup environment (Lucene, Luke, dataset)</td>
</tr>
<tr>
<td>18-Sep-17</td>
<td>2</td>
<td>Basic techniques (P, R, P@10)</td>
<td>2</td>
<td>Text pre-processing, VSM</td>
</tr>
<tr>
<td>25-Sep-17</td>
<td>3</td>
<td>Query processing (P-R, MAP)</td>
<td>3</td>
<td>Evaluation protocols</td>
</tr>
<tr>
<td>02-Oct-17</td>
<td>4</td>
<td>Language models</td>
<td>4</td>
<td>Checkpoint 1: script + visualization (10%)</td>
</tr>
<tr>
<td>09-Oct-17</td>
<td>5</td>
<td>Probabilistic retrieval models</td>
<td>5</td>
<td>Query expansion</td>
</tr>
<tr>
<td>16-Oct-17</td>
<td>6</td>
<td>Information fusion (NDCG)</td>
<td>6</td>
<td>Retrieval models</td>
</tr>
<tr>
<td>23-Oct-17</td>
<td>7</td>
<td>First test (20%)</td>
<td>7</td>
<td>Checkpoint 2: RM systematic comparison (10%)</td>
</tr>
<tr>
<td>30-Oct-17</td>
<td>8</td>
<td>Linked data</td>
<td>8</td>
<td>Indexing multiple fields + L2R</td>
</tr>
<tr>
<td>06-Nov-17</td>
<td>9</td>
<td>Diversity and clustering</td>
<td>9</td>
<td>PageRank</td>
</tr>
<tr>
<td>13-Nov-17</td>
<td>10</td>
<td>Static and dynamic indexing</td>
<td>10</td>
<td>Project</td>
</tr>
<tr>
<td>20-Nov-17</td>
<td>11</td>
<td>Efficient query processing</td>
<td>11</td>
<td>Project</td>
</tr>
<tr>
<td>27-Nov-17</td>
<td>12</td>
<td>Project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>04-Dec-17</td>
<td>13</td>
<td>Project</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11-Dec-17</td>
<td>14</td>
<td>Second test (20%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary

• “Web Search” course context

• Course objectives and plan

• Grading

• Labs