Pre-workshop

- Revise the difference between “Boolean querying” and “ranked querying”. Hypothesise a query where the results would be different under the two modes of querying (treating each as conjunctive).

- Revise “phrase querying”. Issue a couple of multi-word queries to a Web search engine (http://www.google.com) and observe how the results change:
  - Without quotation marks ("")
  - With quotation marks around the entire query
  - With quotation marks around some of the words in the query, but not others
  - With multiple sets of words in quotation marks

- Recall that, for many Information Retrieval problems, we process terms from lowest document frequency, to highest. Why do we do this? Are there any IR tasks where we proceed from highest document frequency to lowest?

- Revise some methods for spelling correction from Knowledge Technologies. In particular, consider “suffix arrays” and “suffix trees” (“tries”). If you aren’t familiar with these concepts, use Google or discuss with other students taking the subject.

- Revise the “vector–space model” and “cosine similarity”. Revise some other metrics for finding similarity between documents in a vector space.

- Revise TF-IDF models and the rationale that goes into each of its components.

Workshop

1. Last week, you found the size of the vocabulary (of tokenised terms) in a document.
   
   (a) What is the size of the vocabulary of “bi–word” terms in a document? How does this depend on the length of the document? How does this relate to the size of the vocabulary of one–word terms? Why does this happen?
   
   (b) Modify your program from last week to find the size of the bi–word vocabulary for one or more documents.
   
   (c) How can you use a bi–word index to evaluate Boolean queries? Is this a good method for performing phrasal querying? What problems might it have?
   
   (d) (Harder) What is the effect of using bi–words rather than single words on the size of an inverted index? (You might find the result surprising!) Why does this happen?

2. What is a “positional index”?
   
   (a) What advantages does a positional index offer over a bi–word index when evaluating phrasal querying? What disadvantages?
(b) Modify your inverted index from last week to incorporate positional information. (Start with a small document collection!)

3. Contrast the methods of “wildcard querying” discussed in the lecture with some other approaches to spelling correction, like `agrep`, “edit distance”, and “tries”.

Post-workshop

- Alter your Boolean retrieval engine from last week to incorporate bi-word indices. Try a few queries of different lengths and observe the results. Are they better or worse than using single word querying?

- (Extension) Modify your Boolean search engine from last week to use a positional index to perform phrasal querying. Observe how the query time and index space changes, as compared to simpler indices and querying methods.

- (Project preparation) Write a program that performs ranked querying using a TF-IDF model in a term vector space, making use of a simple inverted index (incorporating document IDs and term frequencies).