LING 120, Fall 2017
Language and Computers

Instructor: Sowmya Vajjala

Iowa State University, USA

20 September 2017

Class outline

1. Review of last class
2. Introduction to searching the web
3. Issues related to language

Different types of data

1. Structured - Very organized (e.g., a library database - every book has a title, an author, a publisher, other attributes such as number of pages etc.)
2. Unstructured - free-flowing text from which we should extract what we want (e.g., your typical google search)
3. Semi-structured - where the data is generally unstructured, but there are certain patterns we see, which makes it easy to extract content (e.g., if we want to extract all email addresses from a text).

Evaluating Search Results

1. We know how to evaluate search results based on our information need. How do we decide whether a search system is generally good?

Evaluating Search Results

1. We know how to evaluate search results based on our information need. How do we decide whether a search system is generally good?
2. Relevance: Whether the result a search showed us is actually relevant for the user's need.

Evaluating Search Results

1. We know how to evaluate search results based on our information need. How do we decide whether a search system is generally good?
2. Relevance: Whether the result a search showed us is actually relevant for the user's need.
3. Precision: Of all results returned by the search, how many are actually relevant?
4. Recall: Of all the results that are relevant, how many did the search engine manage to retrieve as relevant?
5. The goal of a good search engine is to provide 100% precision and 100% recall.

Precision and Recall: Competing Priorities

1. How do we achieve 100% recall?

Precision and Recall: Competing Priorities

1. How do we achieve 100% recall?
   $\Rightarrow$ Just show up everything in the world - that will automatically achieve 100% recall (How??)

Precision and Recall: Competing Priorities

1. How do we achieve 100% recall?
   $\Rightarrow$ Just show up everything in the world - that will automatically achieve 100% recall (How??)
2. How do we achieve 100% precision?

Precision and Recall: Competing Priorities

1. How do we achieve 100% recall?
   $\Rightarrow$ Just show up everything in the world - that will automatically achieve 100% recall (How??)
2. How do we achieve 100% precision?
   $\Rightarrow$ return that small set of webpages which you are absolutely sure of. (Great, what is the problem then?)

Precision and Recall: Competing Priorities

1. How do we achieve 100% recall?
   $\Rightarrow$ Just show up everything in the world - that will automatically achieve 100% recall (How??)
2. How do we achieve 100% precision?
   $\Rightarrow$ return that small set of webpages which you are absolutely sure of. (Great, what is the problem then?)
3. Often, the goal is to reach a balance between precision and recall.

Last class: A question about "searching"

Work in groups of 2–3 people, think about a solution for this problem, and return your answers to me giving the names of your team members. You can also submit online on Canvas.

Last names in the dictionary

Some words in your dictionary also appear as last names in your phone book. For example, "brooks", "brown", "butler", "hall", and "wright" are in your dictionary, and Brooks, Brown, Butler, Hall, and Wright are all common last names in the U.S.

You would like to make a list of *all* such words. The inefficient way would be to go through the dictionary in order: for each dictionary word, you open the phone book, look up that word, add it to your list if you find it as a last name, and close the phone book again.

(a) Why is it more efficient to keep the phone book open between word look-ups?

(b) What if you have a friend to help you (and two copies of the dictionary and phone book)? How can the two of you divide up the work safely and finish twice as fast?

(c) What if there are three of you instead of two?

source:

http://nacloweb.org/resources/problems/sample/Phonebook.pdf

Answer discussion

http://nacloweb.org/resources/problems/sample/Phonebook-solution.pdf

Search Engines

Typical usage of a search engine

- Let us say we want to search for something ("Iowa State University").
- We go to google and type that string in and choose to search.
- Google returns you lots of results, ranked in some way and paginated.
- We evaluate the results (by looking at the top few results) and seeing if they are relevant

Typical usage of a search engine

- Let us say we want to search for something ("Iowa State University").
- We go to google and type that string in and choose to search.
- Google returns you lots of results, ranked in some way and paginated.
- We evaluate the results (by looking at the top few results) and seeing if they are relevant
- If we are unhappy, we reformulate the query and search repeat the above process again.

Typical usage of a search engine

- Let us say we want to search for something ("Iowa State University").
- We go to google and type that string in and choose to search.
- Google returns you lots of results, ranked in some way and paginated.
- We evaluate the results (by looking at the top few results) and seeing if they are relevant
- If we are unhappy, we reformulate the query and search repeat the above process again.
- From the perspective of google, if we click a result, it can be considered somewhat relevant.

How does a search engine prioritize one page over the other?

Some simple intuitions

- If I am searching for something, and there is a webpage with that "something" in the title, or in URL etc, may be that should be ranked first.
- If there is a Wikipedia page, may be that can show up first.
- If it is a movie, may be the IMDB page can show up first.
- If we are searching for a well-known personality's name and that person has a twitter handle, that should also be seen in top results.
- If there are "advertisements" relevant to my query, they need to be displayed too! (search for, say, "language learning" on google).
- Similar results (or many results from same website) should be grouped together.

Language related issues in a search engine

- Should we consider upper case/lower case differences between words? or should we treat them the same?

Language related issues in a search engine

- Should we consider upper case/lower case differences between words? or should we treat them the same?
- Should I look for exact spellings or may be approximate ones too?

Language related issues in a search engine

- Should we consider upper case/lower case differences between words? or should we treat them the same?
- Should I look for exact spellings or may be approximate ones too? (Tübingen, Tuebingen, Tubingen - all can refer to the same German town).

Language related issues in a search engine

- Should we consider upper case/lower case differences between words? or should we treat them the same?
- Should I look for exact spellings or may be approximate ones too? (Tübingen, Tuebingen, Tubingen - all can refer to the same German town).
- Should I do "stemming", which is basically stripping off word endings? (i.e., car, cars will both be matched). What is the advantage?

Language related issues in a search engine

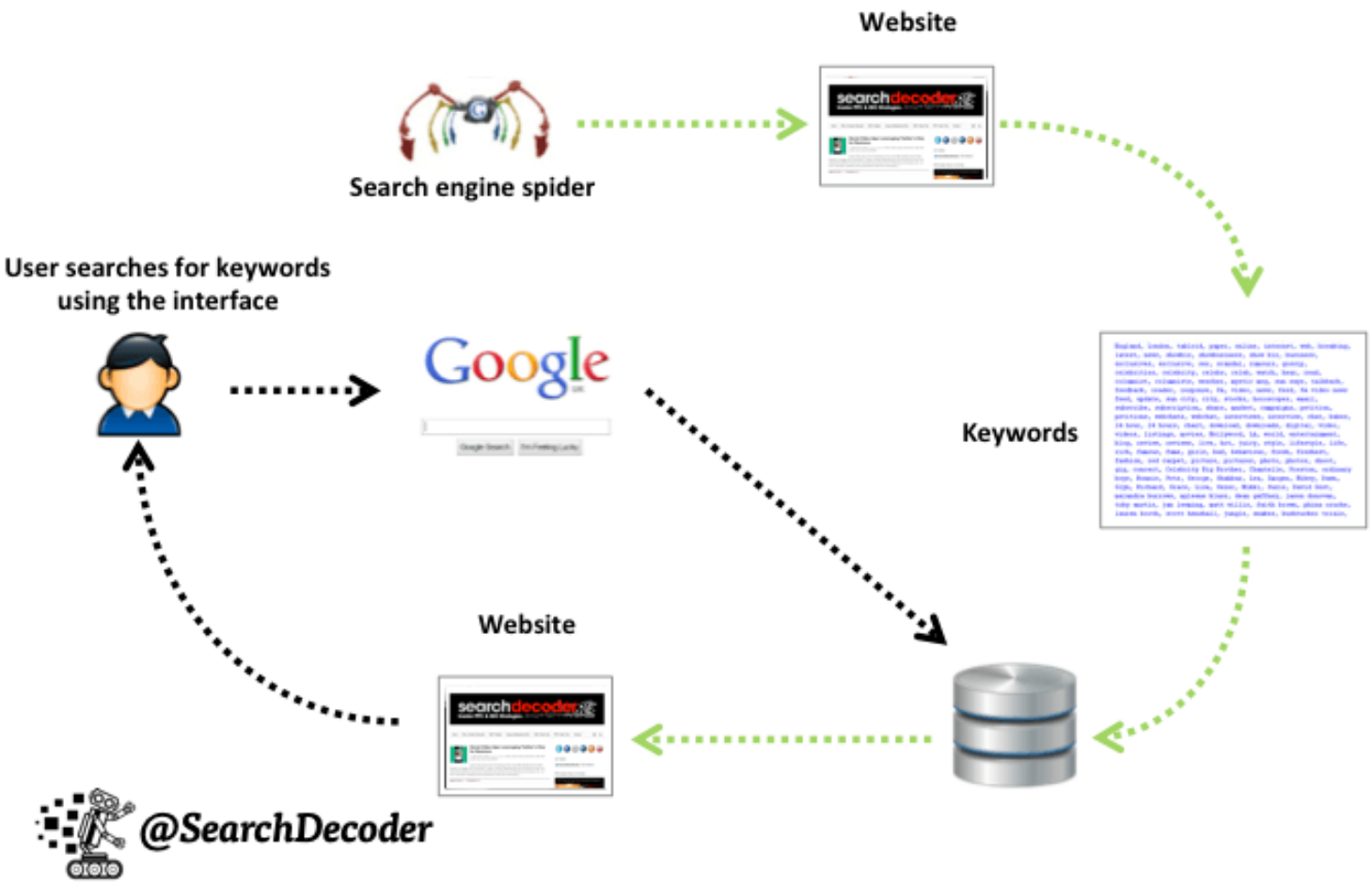
- Should we consider upper case/lower case differences between words? or should we treat them the same?
- Should I look for exact spellings or may be approximate ones too? (Tübingen, Tuebingen, Tubingen - all can refer to the same German town).
- Should I do "stemming", which is basically stripping off word endings? (i.e., car, cars will both be matched). What is the advantage?
- How can I store all information on the web? How can I just search instantly?

Language related issues in a search engine

- Should we consider upper case/lower case differences between words? or should we treat them the same?
- Should I look for exact spellings or may be approximate ones too? (Tübingen, Tuebingen, Tubingen - all can refer to the same German town).
- Should I do "stemming", which is basically stripping off word endings? (i.e., car, cars will both be matched). What is the advantage?
- How can I store all information on the web? How can I just search instantly?
- Ads again: How can I show ads relevant to search query? (Why is it important?)

How does a search engine work?

Which of those require some analysis of language?



source:

https://www.searchdecoder.com/how-do-search-engines-work

Language analysis in search

- "Crawling" to "Indexing": How do we get plain text from webpages? (What is the issue?)
- "Indexing: Purpose is to store all the collected data in an efficient way.
- Understanding a query (may be also translation if it involve cross-language search)
- What snippets need to be shown under a search result?
- Grouping results into categories

Indexing

- One popular indexing structure is: Term-Document Matrix.
- All words as rows, webpages in which they appear as columns.
- Counts or just binary numbers as entries in this matrix.

Indexing

- One popular indexing structure is: Term-Document Matrix.
- All words as rows, webpages in which they appear as columns.
- Counts or just binary numbers as entries in this matrix.
- It is often common to remove "stop words" i.e., removing extremely frequent words such as I, the, a, is etc. (Why?)

Indexing

- One popular indexing structure is: Term-Document Matrix.
- All words as rows, webpages in which they appear as columns.
- Counts or just binary numbers as entries in this matrix.
- It is often common to remove "stop words" i.e., removing extremely frequent words such as I, the, a, is etc. (Why?)
- A more efficient way of representing a TDM is something called inverted index, where you just list the document IDs instead of that big matrix.

TDM and Inverted Index

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Antony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

► **Figure 1.1** A term-document incidence matrix. Matrix element (t, d) is 1 if the play in column d contains the word in row t, and is 0 otherwise.

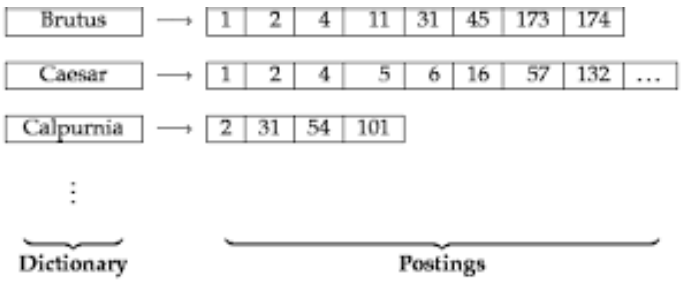
(a) Term Document Matrix

TDM and Inverted Index

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	…
Antony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
…							

► **Figure 1.1** A term-document incidence matrix. Matrix element (t, d) is 1 if the play in column d contains the word in row t, and is 0 otherwise.

(c) Term Document Matrix



► **Figure 1.2** The two parts of an inverted index. The dictionary is commonly kept in memory, with pointers to each postings list, which is stored on disk.

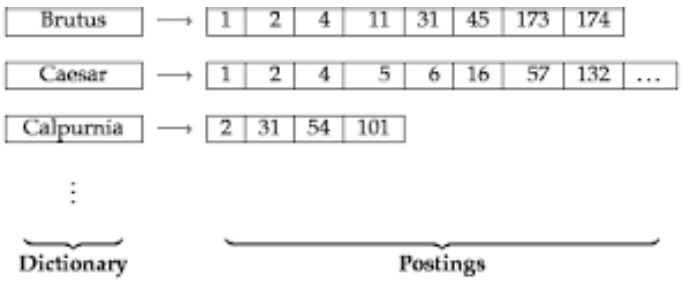
(d) Inverted Index

TDM and Inverted Index

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Antony	1	1	0	0	0	1	
Brutus	1	1	0	1	0	0	
Caesar	1	1	0	1	1	1	
Calpurnia	0	1	0	0	0	0	
Cleopatra	1	0	0	0	0	0	
mercy	1	0	1	1	1	1	
worser	1	0	1	1	1	0	
...							

► **Figure 1.1** A term-document incidence matrix. Matrix element (t, d) is 1 if the play in column d contains the word in row t, and is 0 otherwise.

(e) Term Document Matrix



► **Figure 1.2** The two parts of an inverted index. The dictionary is commonly kept in memory, with pointers to each postings list, which is stored on disk.

(f) Inverted Index

- TDM for WWW is incredibly huge.
- Modern search engines use several other characteristics as well (beyond words and phrases).

How is this useful when someone searches?

If I search for "Iowa" AND "state" AND "university" - I look for documents that contain all three words and show these!

Small Question

3. **ALL**: Imagine that there are 11 books in a library with the following subject fields:

1) rock paper
2) rock scissors
3) rock bomb
4) paper scissors
5) paper bomb
6) scissors bomb
7) rock paper scissors
8) rock paper bomb
9) rock scissors bomb
10) paper scissors bomb
11) rock paper scissors bomb

Looking at the list of queries below, which of the subject numbers do they match? For example, rock AND bomb matches 3, 8, 9, and 11.

(a) rock OR paper OR scissors
(b) rock AND (paper OR scissors)
(c) (rock AND paper) OR (scissors AND bomb)
(d) (rock OR paper) AND (scissors OR bomb)
(e) rock AND (paper OR (scissors AND bomb))
(f) ((rock AND paper) OR scissors) AND bomb

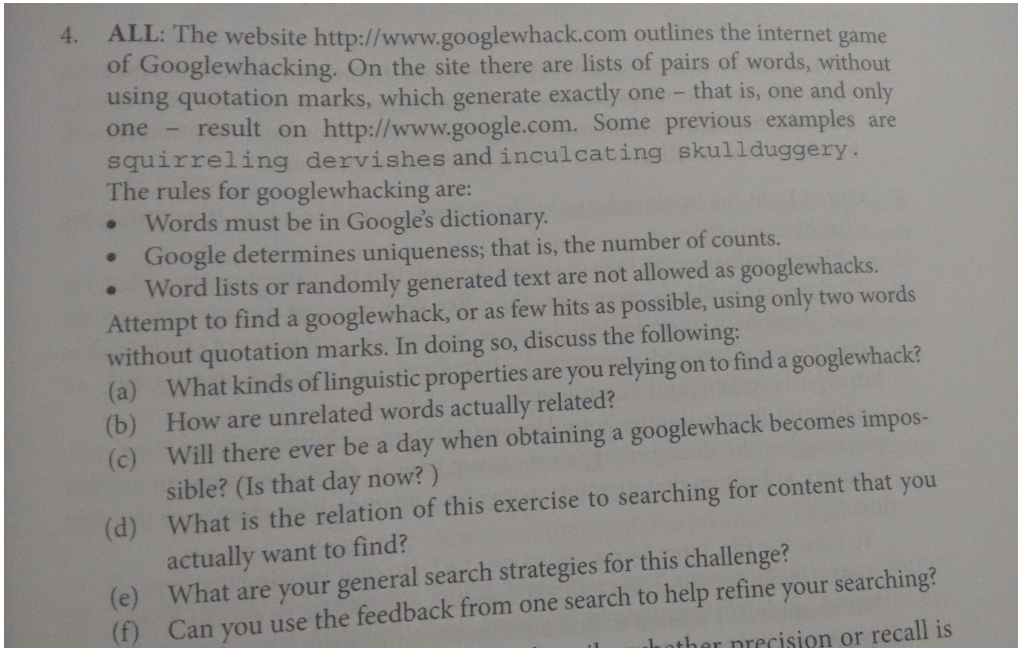
Questions to resolve by Friday

- Okay, indexing is cool. But for any query, I may still end up with 10000 results. How can I rank them?
- How do I ensure only good quality pages get ranked on top.

Attendance Exercise(s) - Lots of questions

Think about these problems and submit your thoughts online on Canvas forum for today.

- Is "popularity" a good heuristic to rank a page? Provide one example where your search query results in a popular page as the top result, but is incorrect for your need. What is the rank of the page that met your need?
- Question on Googlewhack. That website in the picture does not work though. You should use other means to know more.



4. **ALL**: The website http://www.googlewhack.com outlines the internet game of Googlewhacking. On the site there are lists of pairs of words, without using quotation marks, which generate exactly one – that is, one and only one – result on http://www.google.com. Some previous examples are `squirreling dervishes` and `inculcating skullduggery`.

The rules for googlewhacking are:

- Words must be in Google's dictionary.
- Google determines uniqueness; that is, the number of counts.
- Word lists or randomly generated text are not allowed as googlewhacks.

Attempt to find a googlewhack, or as few hits as possible, using only two words without quotation marks. In doing so, discuss the following:

(a) What kinds of linguistic properties are you relying on to find a googlewhack?

(b) How are unrelated words actually related?

(c) Will there ever be a day when obtaining a googlewhack becomes impossible? (Is that day now?)

(d) What is the relation of this exercise to searching for content that you actually want to find?

(e) What are your general search strategies for this challenge?

(f) Can you use the feedback from one search to help refine your searching?

... whether precision or recall is