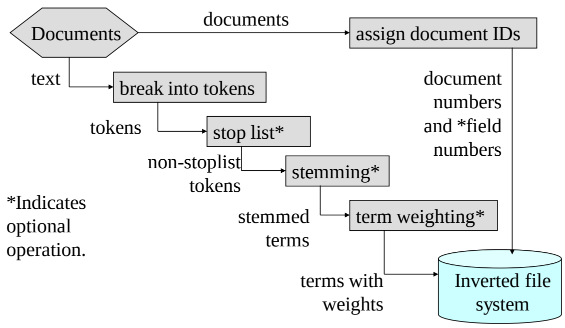
Indexation

1

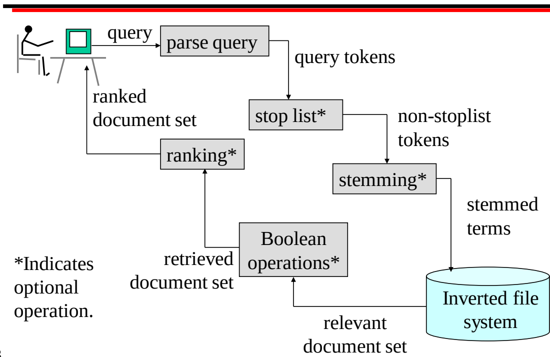
Indexing Subsystem



*Indicates optional operation.

2

Search Subsystem



*Indicates optional operation.

3

Decisions in Building the Index: What is a Document?

- For a compound document, is each part indexed separately, e.g., an email message with attachments?
- Is a long item divided into several mini-documents, e.g., book chapters?

Several of the examples in the next few slides are based on Manning et al., chapter 2.

4

Lexical Analysis: Term

What is a term?

Free text indexing

A term is a group of characters, derived from the input string, that has some collective significance, e.g., a complete word.

Usually, terms are strings of letters, digits or other specified characters, separated by punctuation, spaces, etc.

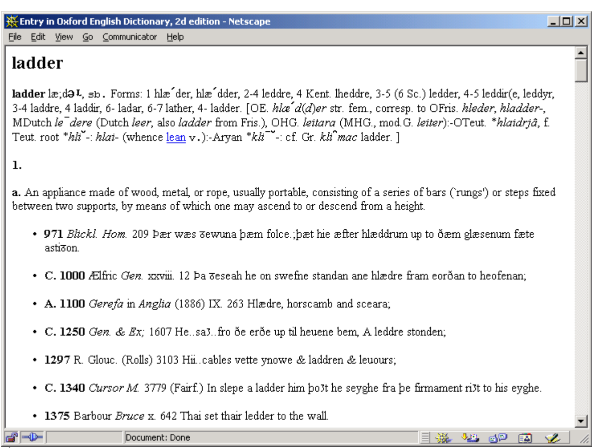
5

Lexical Analysis: Tokens and Index Terms

A **token** is a strong of characters extracted from a document, e.g.,

The discussion classes on Wednesday evenings are from 7:30 to 8:30 p.m.

6


Entry in Oxford English Dictionary, 2d edition - Netscape

File Edit View Go Communicator Help

ladder

ladder læ;dəɹ, sb. Forms: 1 hlæ´der, hlæ´dder, 2-4 leddre, 4 Kent. lheddre, 3-5 (6 Sc.) ledder, 4-5 leddir(e, leddyr, 3-4 laddre, 4 laddir, 6- ladar, 6-7 lather, 4- ladder. [OE. *hlæ´d(d)er* str. fem., corresp. to OFris. *hleder, hladder-*, MDutch *le¯dere* (Dutch *leer*, also *ladder* from Fris.), OHG. *leitara* (MHG., mod.G. *leiter*):-OTeut. **hlaidrjâ*, f. Teut. root **hli˘*-: *hlai-* (whence lean v.):-Aryan **kli¯˘*-: cf. Gr. *kli^mac* ladder.]

1.

a. An appliance made of wood, metal, or rope, usually portable, consisting of a series of bars (`rungs') or steps fixed between two supports, by means of which one may ascend to or descend from a height.

- **971** *Blickl. Hom.* 209 Þær wæs ʒewuna þæm folce.;þæt hie æfter hlæddrum up to ðæm glæsenum fæte astiʒon.
- **C. 1000** Ælfric *Gen.* xxviii. 12 Þa ʒeseah he on swefne standan ane hlædre fram eorðan to heofenan;
- **A. 1100** *Gerefa* in *Anglia* (1886) IX. 263 Hlædre, horscamb and sceara;
- **C. 1250** *Gen. & Ex;* 1607 He..saȝ..fro ðe erðe up til heuene bem, A leddre stonden;
- **1297** R. Glouc. (Rolls) 3103 Hii..cables vette ynowe & laddren & leuours;
- **C. 1340** *Cursor M.* 3779 (Fairf.) In slepe a ladder him þoȝt he seyghe fra þe firmament riȝt to his eyghe.
- **1375** Barbour *Bruce* x. 642 Thai set thair ledder to the wall.

Document: Done

Decisions in Building the Index: What is a Term?

- Underlying character set, e.g., printable ASCII, Unicode, UTF8.
- Special formats, e.g., *doc* or *html* (e.g., * *)
- Is there a controlled vocabulary? If so, what words are included?
- List of stopwords.
- Rules to decide the beginning and end of words, e.g., spaces or punctuation.
- Character sequences not to be indexed, e.g., very short terms, sequences of numbers.

8

Lexical Analysis: Tokens and Index Terms

In full text indexing, an **index term** is an equivalence class of **tokens**, with some tokens rejected. Token normalization is the set of **rules** that map tokens into equivalence classes. Even within the English language there are numerous decisions to be made.

- Case-folding: Map all letters to upper case (but *Windows* maps to *windows*)
- Accents and diacritics: Ignore (usually OK for English)
- Abbreviations: If *U.S.A. ~ usa*, is *C.A.T. ~ cat*?
- Dates: Can we map *16 August 1997* to *8/16/97*?
- Versions of English: There are numerous versions of English, e.g., British and American English

9

Lexical Analysis: Tokens and Index Terms

Here are some more examples from Manning, et al.

Apostrophe:

Is boys' ~ boys?

Is O'Neill ~ ONeill?

Special tokens

M*A*S*H

C++

http://www.infosci.cornell.edu/courses/info430/2007fa/

10

Lexical Analysis: Choices

Punctuation: In technical contexts, punctuation may be used as a character within a term, e.g., *wordlist.txt*.

Hyphens: Which of the following rules is most useful?

(a) Treat as separators: *state-of-art* is treated as *state of art*.

(b) Ignore: *on-line* is treated as *online*.

(c) Retain: *Knuth-Morris-Pratt Algorithm* is unchanged.

Digits: Most numbers do not make good terms, but some are parts of proper nouns or technical terms: *CS430*, *Opus 22*.

11

Lexical Analysis: Choices

The modern tendency, for ***free text searching***, is to map upper and lower case letters together in index terms, but otherwise to minimize the changes made at the lexical analysis stage.

With ***controlled vocabulary***, the lexical decisions are made in creating the vocabulary.

12

Stop Lists

Very common words, such as *of*, *and*, *the*, are rarely of use in information retrieval.

A **stop list** is a list of such words that are removed during lexical analysis.

A long stop list saves space in indexes, speeds processing, and eliminates many false hits.

However, common words are sometimes significant in information retrieval, which is an argument for a short stop list. (Consider the query, "To be or not to be?")

13

Example: Stop List for Assignment 1

a	about	an	and
are	as	at	be
but	by	for	from
has	have	he	his
in	is	it	its
more	new	of	on
one	or	said	say
that	the	their	they
this	to	was	who
which	will	with	you

14

Example: the WAIS stop list (first 84 of 363 multi-letter words)

about	above	according	across	actually	adj
after	afterwards	again	against	all	almost
alone	along	already	also	although	always
among	amongst	an	another	any	anyhow
anyone	anything	anywhere	are	aren't	around
at	be	became	because	become	becomes
becoming	been	before	beforehand	begin	beginning
behind	being	below	beside	besides	between
beyond	billion	both	but	by	can
can't	cannot	caption	co	could	couldn't
did	didn't	do	does	doesn't	don't
down	during	each	eg	eight	eighty
either	else	elsewhere	end	ending	enough
etc	even	ever	every	everyone	everything

15

Problems with Stop Words

Languages

Multi-lingual document collections have special problems, e.g., *die* is a very common word in German but less common in English.

Semantic information

Prepositions and other common words may be important in a search, e.g., *President* of the *United States*

Queries

Some queries are entirely stop words, e.g., *To be or not to be*

16

Suggestions for Including Words in a Stop List

- **Include** the most common words in the English language (perhaps 10 to 250 words).
- **Do not include** words that might be important for retrieval (Among the 200 most frequently occurring words in general literature in English are *time*, *war*, *home*, *life*, *water*, and *world*).
- In addition, **include** words that are very common in context (e.g., *computer*, *information*, *system* in a set of computing documents).

17

Stop list policies

How many words should be in the stop list?

- Long list lowers recall but increase precision

There is very little systematic evidence to use in selecting a stop list.

18

Stop Lists in Practice

The modern tendency is:

(a) have very short stop lists for broad-ranging or multi-lingual document collections, especially when the users are not trained.

(b) have longer stop lists for document collections in well-defined fields, especially when the users are trained professional.

19

Lemmatization

- Reduce inflectional/variant forms to base form
- E.g.,
  - *am, are, is* → *be*
  - *car, cars, car's, cars'* → *car*
- *the boy's cars are different colors* → *the boy car be different color*
- Lemmatization implies doing "proper" reduction to dictionary headword form

20

Stemming

Morphological variants of a word (morphemes). Similar terms derived from a common stem:

engineer, engineered, engineering
use, user, users, used, using

Stemming in Information Retrieval. Words with a common stem and mapped into the same index term.

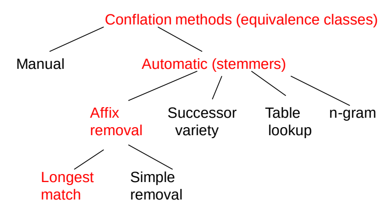
For example, *read*, *reads*, *reading*, and *readable* are mapped onto the index term *read*.

Stemming consists of removing **suffixes** and **conflating** the resulting morphemes. Occasionally, **prefixes** are also removed.

21

Categories of Stemmer

The following diagram illustrate the various categories of stemmer. Porter's algorithm is shown by the red path.



22

Porter Stemmer

A multi-step, longest-match stemmer.

M. F. Porter, An algorithm for suffix stripping. (Originally published in *Program*, 14 no. 3, pp 130-137, July 1980.)
http://www.tartarus.org/~martin/PorterStemmer/def.txt

Notation

___v	vowel(s)
c	constant(s)
$(vc)_m$	vowel(s) followed by constant(s), repeated m times

The stem of any word can be written: $[c](vc)_m[v]$

m is called the measure of the word

23

Porter's Stemmer

Multi-Step Stemming Algorithm

Complex suffixes

Complex suffixes are removed bit by bit in the different steps. Thus:

GENERALIZATIONS

becomes GENERALIZATION (Step 1)
becomes GENERALIZE (Step 2)
becomes GENERAL (Step 3)
becomes GENER (Step 4)

[In this example, note that Steps 3 and 4 appear to be unhelpful for information retrieval.]

24

Porter Stemmer: Example Step 1a

Suffix	Replacement	Examples
sses	ss	caresses -> caress
ies	i	ponies -> poni ties -> ti
ss	ss	caress -> caress
s		cats -> cat

At each step, carry out the **longest match** only.

25

Porter Stemmer: Example Step 1b

Conditions	Suffix	Replacement	Examples
(m > 0)	eed	ee	feed -> feed agreed -> agree
(*v*)	ed	*null*	plastered -> plaster bled -> bled
(*v*)	ing	*null*	motoring -> motor sing -> sing

Notation

m - the measure of the stem

v - the stem contains a vowel

26

Porter Stemmer: Example Step 5a

Some of the steps are based on peculiarities of English, e.g.,

(m>1)	e ->	probate -> probat rate -> rate
(m=1 and not *o)	e ->	cease -> ceas

*o - the stem ends cvc, where the second c is not w, x or y (e.g. -wil, -hop).

27

Porter Stemmer: Experimental Results

Suffix stripping of a vocabulary of 10,000 words

Number of words reduced in step 1: 3597
step 2: 766
step 3: 327
step 4: 2424
step 5: 1373
Number of words not reduced: 3650

The resulting vocabulary of stems contained 6370 distinct entries. Thus the suffix stripping process reduced the size of the vocabulary by about one third.

28

Stemming in Practice

Evaluation studies have found that stemming can affect retrieval performance, usually for the better, but the results are mixed.

- Effectiveness is dependent on the vocabulary. Fine distinctions may be lost through stemming.
- Automatic stemming is as effective as manual conflation.
- Performance of various algorithms is similar.

Porter's Algorithm is entirely empirical, but has proved to be an effective algorithm for stemming English text when the users of the search system are experienced searchers.

29

Selection of tokens, weights, stop lists and stemming

Special purpose collections (e.g., law, medicine, monographs)

Best results are obtained by tuning the search engine for the characteristics of the collections and the expected queries.

It is valuable to use a **training set** of queries, with lists of relevant documents, to tune the system for each application.

General purpose collections (e.g., news articles)

The modern practice is to use a basic weighting scheme (e.g., *tf.idf*), a simple definition of token, a short stop list and little stemming except for plurals, with minimal conflation.

Web searching combine similarity ranking with ranking based on document importance.

30