Text Operations

Uke 40 – Lecture 7

Aim of this lecture

- Learn about text operation techniques

Objectives

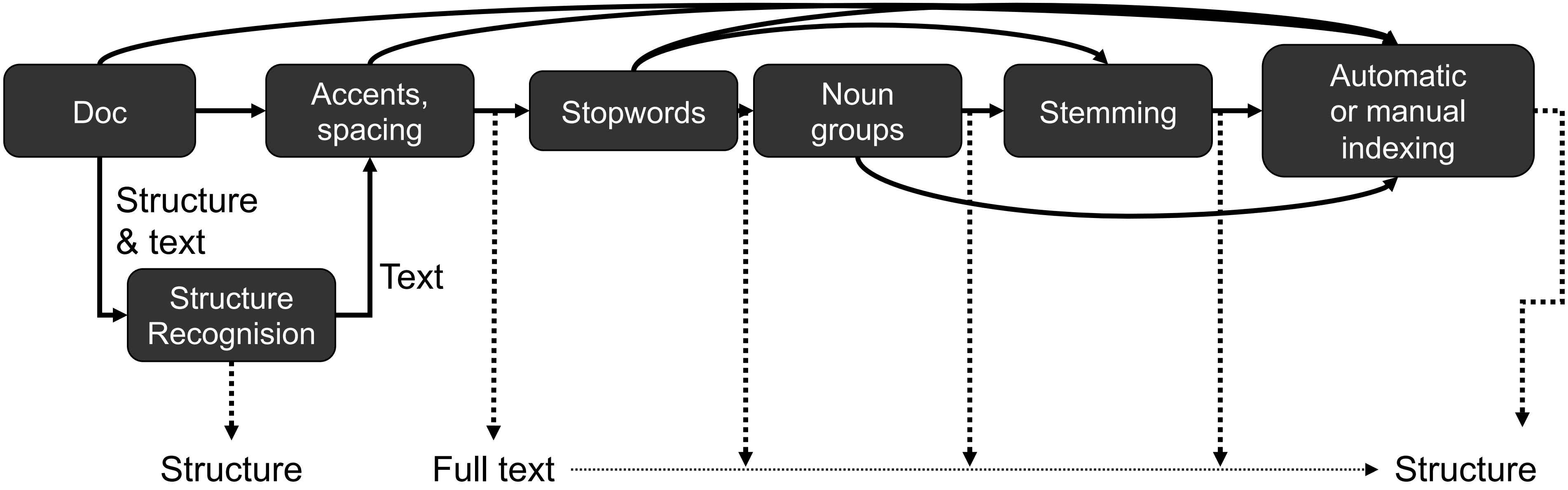
- Know about central methods for document preprocessing procedures
  - **What, why, how.**
- Learn about text compression methods/models
  - What, why and how
  - Know to compare existing models

Content

- Document Preprocessing
  1. **Lexical Analysis**
  2. **Stopwords Elimination**
  3. **Stemming**
  4. **Index Term Selection**
  5. **Thesauri**

- Text Compression
  - Statistical Methods
  - Dictionary Methods
  - Inverted File Compression

Logical view of a document



Document Preprocessing

- Five text transformations (operations)

1. **Lexical Analysis**
   - Level of handling digits, hyphens, punctuation marks, case of letters
2. **Stopwords Elimination**
   - Filter out words with low discrimination
3. **Stemming**
   - Remove prefixes and suffixes
   - Allow queries to have syntactic variations

Document Preprocessing (2)

4. **Index Term Selection**
   - Noun words carry more semantics than adjectives, adverbs, and verbs
5. **Thesauri**
   - Term categorization structure
   - Used in query expansion

Lexical Analysis

- Converting characters to into words:
  - Handling digits, hyphens, punctuation marks, case of letters
- Digits
  - Numbers are vague by themselves
    - Q: number of deaths due to car accidents between 1910 and 1989
      - 1910, 1989 are not appropriate for index terms
  - Must be careful when mixed with words
    - e.g., 510B.C, 16 digits credit card number
  - Date and number normalization

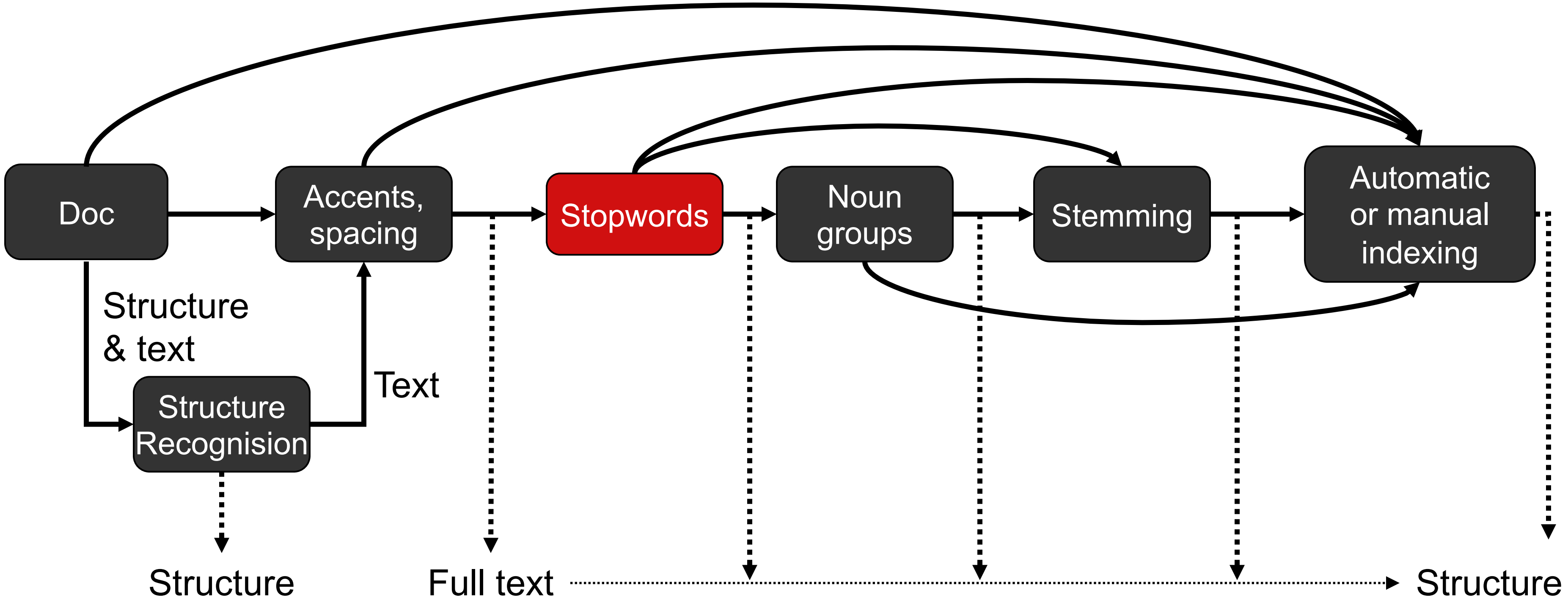
Lexical Analysis (2)

- Hyphens
  - Break up (popular approach)
    - e.g., ‘`state-of-the-art`’ → ‘`state of the art`’
  - Keep
    - e.g., ‘`B-49`’

Lexical Analysis (3)

- Punctuation marks
  - Remove (popular approach)
    - e.g., D: ‘510B.C.’ → ‘510BC’, Q: ‘510B.C’ → ‘510BC’
  - Keep
    - e.g., program code ‘x.id’
- Case of letters
  - Convert all the texts to either lower or upper case
  - Do not convert
    - e.g., Unix commands

Logical view of a document



Stopwords Elimination

- About 80% of occurrences are ***useless for retrieval***
- Stopwords
  - Articles, prepositions, conjunctions
  - Some verbs, adverbs, adjectives
  - Some source define a list of stopwords (e.g., 425)
- Can reduce the size of indexes by 40%
- Disadvantage
  - Might reduce recall
    - e.g., 'to be or not to be' → 'be'
    - Some search engine adopt full text index

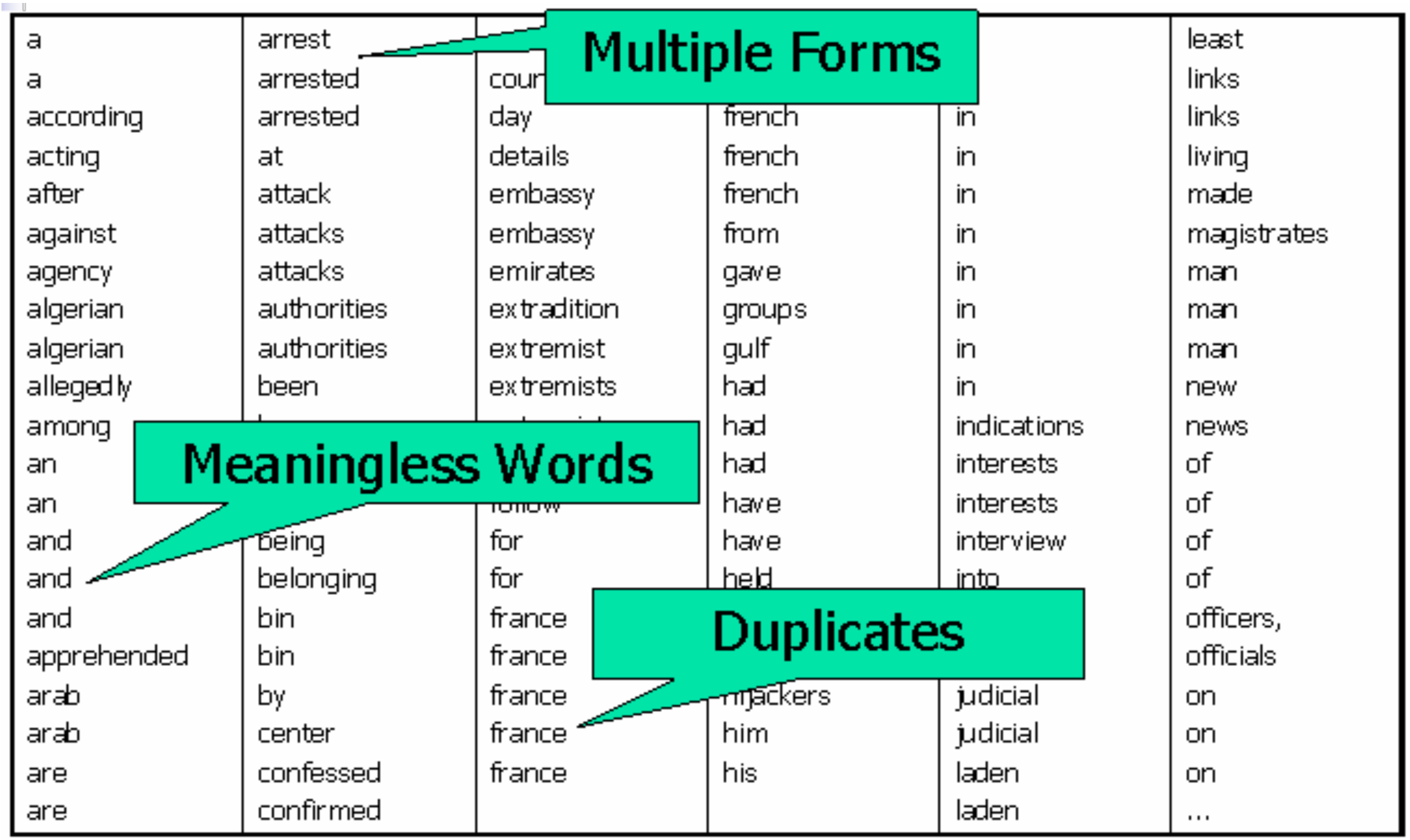
Example document

French arrest suspected extremists
PARIS, France -- Eight people suspected of belonging to extremist Islamic groups have been arrested in France.

They were apprehended by counter-intelligence officers, acting
on orders of magistrates probing terrorist threats made against U.S
. interests in France. According to French authorities, the
U.S. Embassy in Paris was among the possible targets.

French authorities opened a probe into whether U.S. interests in Fr
ance were under threat from attacks the day before suicide
hijackers smashed planes into New York's World Trade Center and
the Pentagon in Washington.

Partial word list



a
a
according
acting
after
against
agency
algerian
algerian
allegedly
among
an
an
and
and
and
apprehended
arab
arab
are
are

arrest
arrested
arrested
at
attack
attacks
attacks
authorities
authorities
been
being
belonging
bin
bin
by
center
confessed
confirmed

cour
day
details
embassy
embassy
emirates
extradition
extremist
extremists
follow
for
for
france
france
france
france
france

french
french
french
from
gave
groups
gulf
had
had
had
have
have
held
hijackers
him
his

in
in
in
in
in
in
in
in
indications
interests
interests
interview
into
judicial
judicial
laden
laden

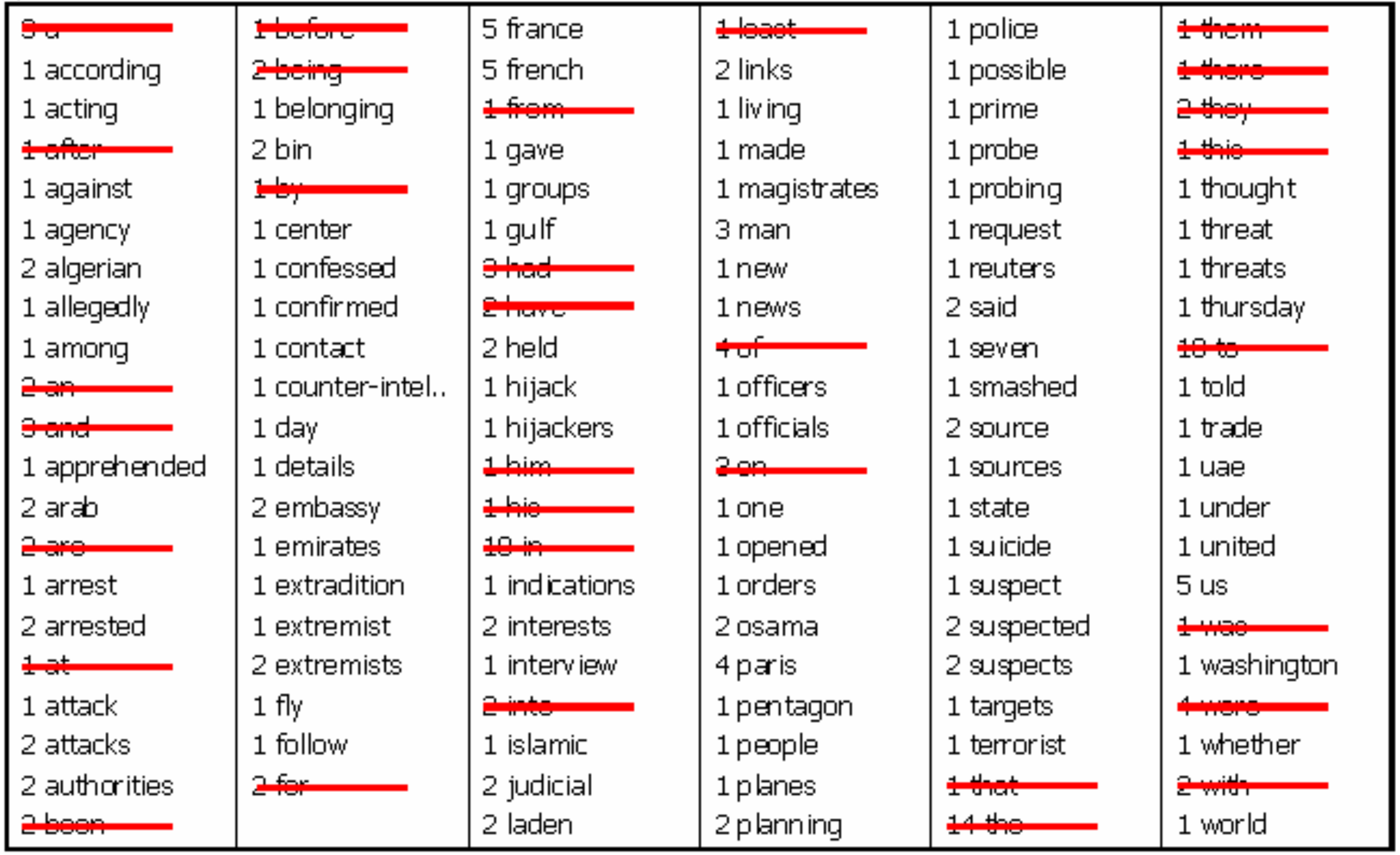
least
links
links
living
made
magistrates
man
man
man
new
news
of
of
of
of
officers,
officials
on
on
on
...

Multiple Forms

Meaningless Words

Duplicates

Stopword removal



~~9 a~~
1 according
1 acting
~~1 after~~
1 against
1 agency
2 algerian
1 allegedly
1 among
~~2 an~~
~~3 and~~
1 apprehended
2 arab
~~2 are~~
1 arrest
2 arrested
~~1 at~~
1 attack
2 attacks
2 authorities
~~2 been~~

~~1 before~~
~~2 being~~
1 belonging
2 bin
~~1 by~~
1 center
1 confessed
1 confirmed
1 contact
1 counter-intel..
1 day
1 details
2 embassy
1 emirates
1 extradition
1 extremist
2 extremists
1 fly
1 follow
~~2 for~~

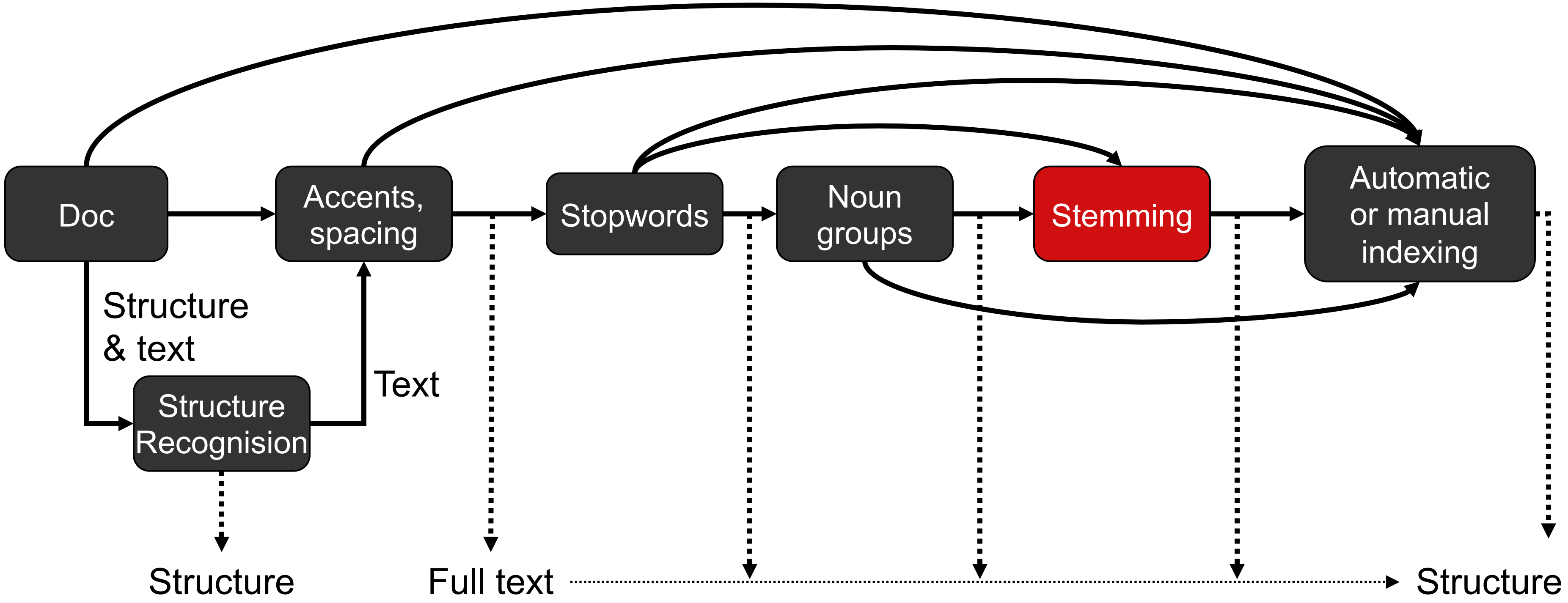
5 france
5 french
~~1 from~~
1 gave
1 groups
1 gulf
~~3 had~~
~~2 have~~
2 held
1 hijack
1 hijackers
~~1 him~~
~~1 his~~
~~10 in~~
1 indications
2 interests
1 interview
~~2 into~~
1 islamic
2 judicial
2 laden

~~1 least~~
2 links
1 living
1 made
1 magistrates
3 man
1 new
1 news
~~4 of~~
1 officers
1 officials
~~2 on~~
1 one
1 opened
1 orders
2 osama
4 paris
1 pentagon
1 people
1 planes
2 planning

1 police
1 possible
1 prime
1 probe
1 probing
1 request
1 reuters
2 said
1 seven
1 smashed
2 source
1 sources
1 state
1 suicide
1 suspect
2 suspected
2 suspects
1 targets
1 terrorist
~~1 that~~
~~14 the~~

~~1 them~~
~~1 there~~
~~2 they~~
~~1 this~~
1 thought
1 threat
1 threats
1 thursday
~~10 to~~
1 told
1 trade
1 uae
1 under
1 united
5 us
~~1 was~~
1 washington
~~4 were~~
1 whether
~~2 with~~
1 world

Logical view of a document



Stemming

- Stem
  - Portion of a word after removing affixes (prefixes and suffixes)
  - e.g., stem: connect, variants: connected, connecting, connection, connections
- Advantages
  - Improve retrieval performance
  - Reduce index size
- Controversy
  - Frakes experiment did not conclude the benefit of stemming
  - Many search engines do not adopt stemming

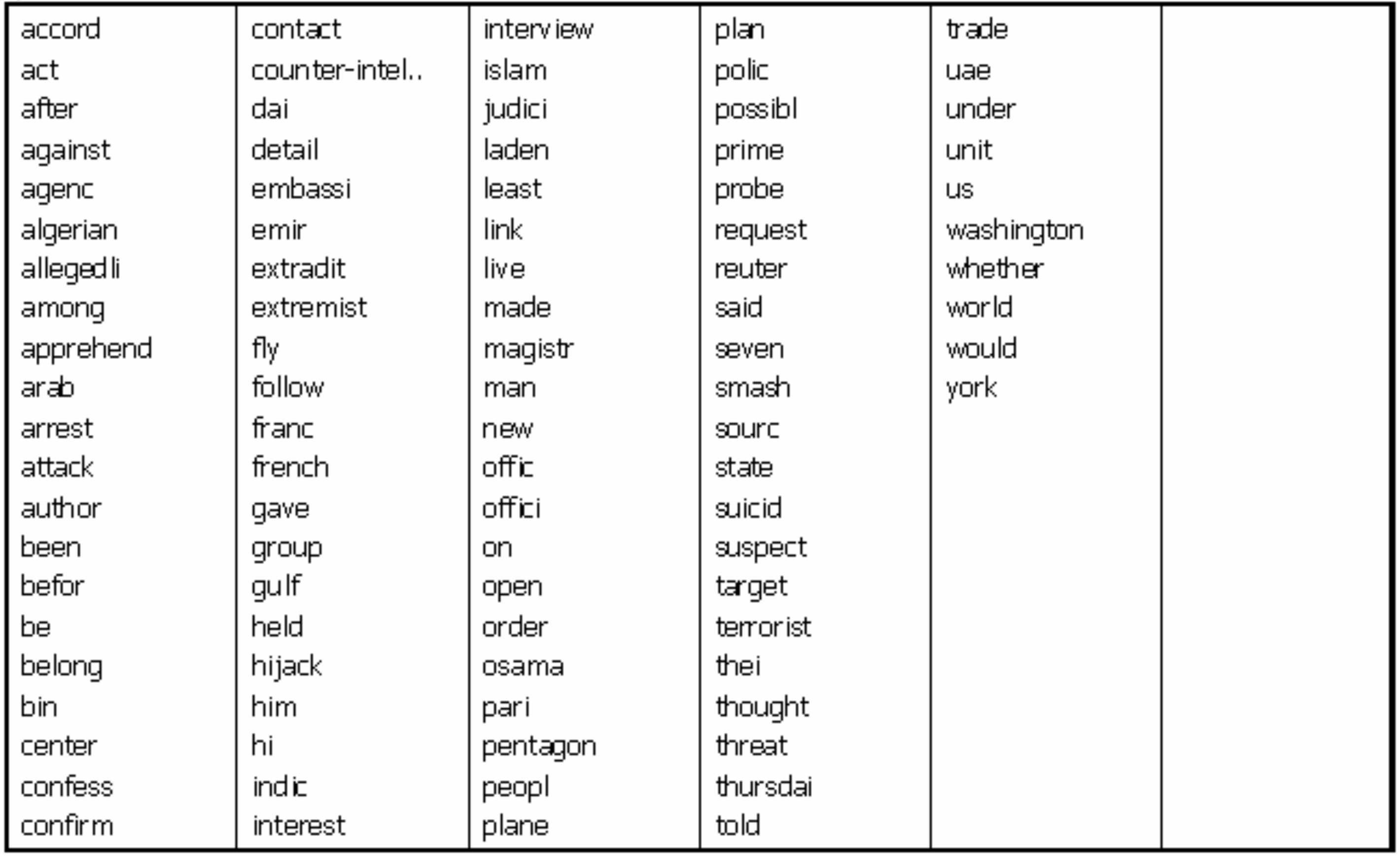
Stemming (2)

- Four types of stemming strategies
  - **Affix removal**
    - Simple, intuitive
  - **Table lookup**
    - Look for the stem of a word in a table
    - Need big storage space for the table
  - **Successor variety**
    - Determine morpheme boundaries
    - Knowledge from linguistics
  - **N-grams**
    - Identification of bigrams, trigrams, etc.
    - Clustering procedure rather than stemming

Stemming (3)

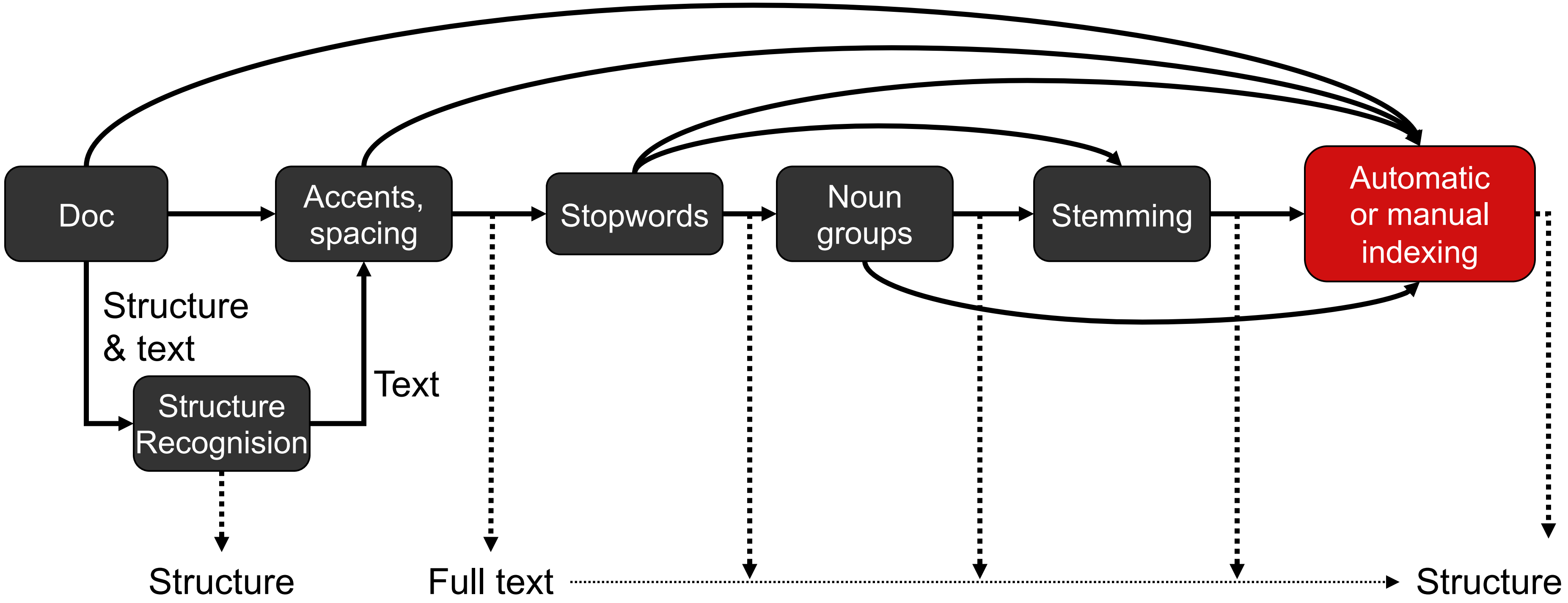
- Affix removal (suffix removal)
  - Porter algorithm
    - Simple, elegant, popular
    - Use a suffix list
    - Apply series of rules to the suffixes
    - Look for longest suffix
    - Example
      - Rules: sses → ss, s → ε
      - stresses → stress

Stemming



accord
act
after
against
agenc
algerian
allegedli
among
apprehend
arab
arrest
attack
author
been
befor
be
belong
bin
center
confess
confirm
contact
counter-intel..
dai
detail
embassi
emir
extradit
extremist
fly
follow
franc
french
gave
group
gulf
held
hijack
him
hi
indic
interest
interview
islam
judici
laden
least
link
live
made
magistr
man
new
offic
offici
on
open
order
osama
pari
pentagon
peopl
plane
plan
polic
possibl
prime
probe
request
reuter
said
seven
smash
sourc
state
suicid
suspect
target
terrorist
thei
thought
threat
thursdai
told
trade
uae
under
unit
us
washington
whether
world
would
york

Logical view of a document



Index Term Selection

- Two approaches
  - Manual selection by specialists
  - Automatic selection
    - Identify noun groups

- Automatic selection
  - Systematic elimination of verbs, adjectives, adverbs, connectives, pronouns except nouns
    - Nouns carry more semantic meaning than others
  - Combine two or three nouns in a single indexing component (concept)
    - e.g., “computer science”
    - Noun group: a set of nouns whose syntactic distance < threshold

Thesauri

- Consists of
  - List of important words in a given domain
  - For each word, set of related words
    - Derived from synonymity relationship
  - Also include some structures
    - Peter Roget’ s thesaurus (generic)
      - Organize words and phrases in categories and subcategories
      - e.g., associate synonyms

Corwardly adjective
Ignobly lacking in courage: cowardly turncoats.
Syns: chicken (slang), chicken-hearted, craven, dastardly,
Faint-hearted, gutless, lily-livered, pusillanimous, unmanly, yellow (slang)
Yellow-bellied (slang)

- Thesaurus can be specific to certain domain
  - e.g., The Thesaurus of Engineering and Scientific Terms

Thesauri (2)

- Main purpose of thesauri
  - Provide standard vocabulary
  - Assist user for proper query terms
  - Provide some hierarchy that allows the broadening or narrowing the query
  - Use controlled vocabulary for indexing and searching
  - Retrieval based on concepts rather than on words (toward semantic)
    - Especially useful in a specific domain like medical domain
- Is the thesauri advantages for Web?
  - Yes. e.g, Yahoo (provides term classification hierarchy)
  - No. e.g, most of search engine use all words as index terms

Thesaurus Index Terms

- Concept
  - Basic semantic unit
  - Individual words, group of words, phrases
  - Adjective + noun, e.g., ‘ballistic missiles’
    - Want to index ‘missiles’ instead of ‘ballistic’
    - Change the order: ‘missiles, ballistic’
- Need definition or explanation
  - for each term for precise meaning
    - e.g. ‘seal’ : fasten vs. establish

Thesaurus Index Terms (2)

- Thesaurus term relationships
  - Synonyms, near-synonyms
  - Co-occurrence within documents
    - e.g., similarity thesaurus, statistical thesaurus
  - Broader terms (BT) or narrower terms (NT)
  - Related terms (RT)
- Where to use?
  - Query expansion

Text Compression

- Compression models
  - **Static**
    - **Model the distribution once**, use over and over again
    - Disadvantage: poor performance when model and the actual data have different distribution
      - E.g., English literature text vs. financial text

Text Compression (2)

- Compression models
  - **Adaptive**
    - **Progressively learn** about statistical distribution of the texts
    - Need one pass over the text
    - Advantage: good for **general purpose**
    - Disadvantage: decompression start from the beginning
      - *Cannot randomly access* the compressed patterns

Text Compression (3)

- Compression models (cont’ d)
  - **Semi-static**
    - Two passes: First pass for modeling, second pass for for compress
    - Disadvantages: Two passes. Model should be stored and sent to decoder
    - Advantage: **Direct access => Good for IR**

Text Compression (2)

- Compression models
  - Character based
  - Word based
    - Treat words as symbols
    - Advantages
      - Much better compression rate
      - Words are the units of texts

Statistical Coding

- Main goal
  - assign short codes to likely symbols and long codes to unlikely ones
- issues
  - compression ratio
  - encode/decode speed

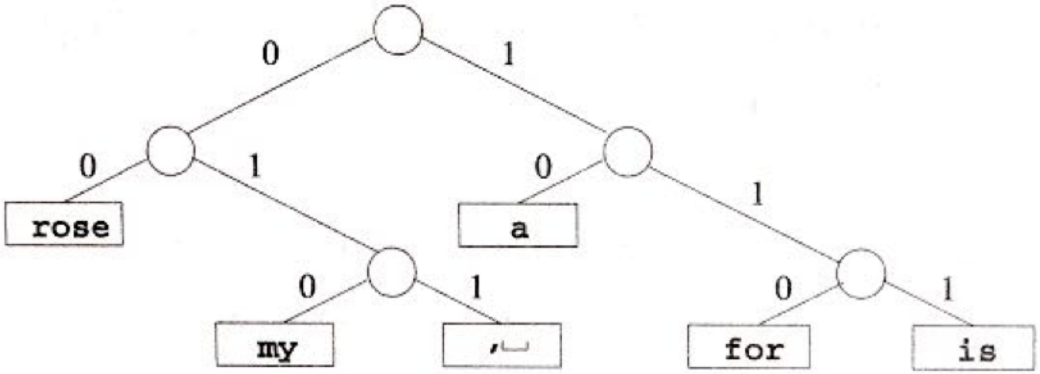
Statistical Coding (2)

- **Huffman coding**
  - can be static, semi-static, adaptive
  - decompression can start in the middle (static, semi-static)

Huffman Coding

`for each rose, a rose is a rose`

- Example
  - symbols: { ',' , a, each, for, is, rose}
  - frequencies: 1, 2, 1, 1, 1, 3
  - Huffman coding tree
  - decompression
    - traverse the compressed code and the Huffman tree together
    - whenever a leaf is reached, output the corresponding symbol



Original text: `for my rose, a rose is a rose`

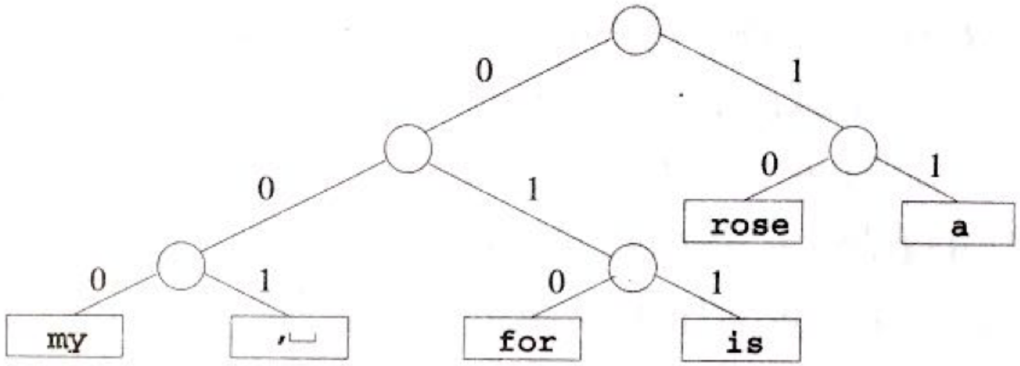
Compressed text: `110 010 00 011 10 00 111 10 00`

Huffman Coding (2)

- How to construct Huffman tree
  - merge the two smallest frequent symbols to make a node
    - associate the combined frequency to the node
  - repeat till we run of out symbols
  - number of distinct Huffman trees for a given problem?
    - Many
- canonical tree
  - Impose some order
  - Height of left subtree of any node is >= height of right subtree
  - All leaves are in increasing order of probabilities from left to right

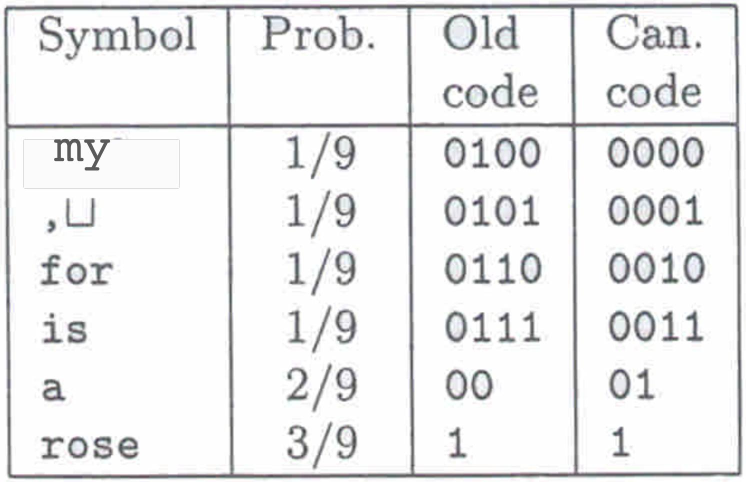
Huffman Coding (3)

- Canonical form



Original text: for my rose, a rose is a rose

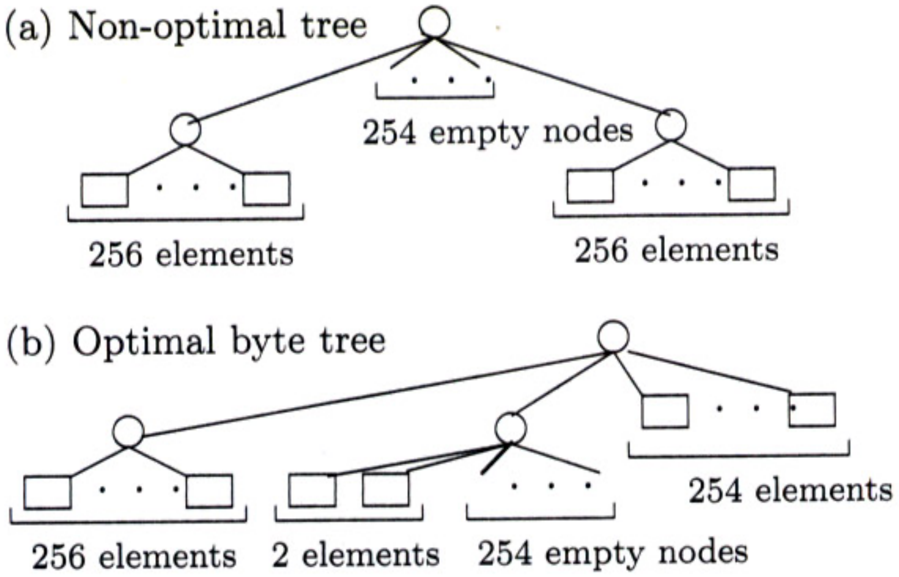
Compressed text: 010 000 10 001 11 10 011 11 10



Symbol	Prob.	Old code	Can. code
my	1/9	0100	0000
,⊔	1/9	0101	0001
for	1/9	0110	0010
is	1/9	0111	0011
a	2/9	00	01
rose	3/9	1	1

Byte-Oriented Huffman Code

- Bit code is extended to byte
- The tree has degree 256 instead of 2
- typically symbols are represented by <= 5 bytes
  - e.g., rose = ‘47 131 8’



Byte-Oriented Huffman Code (2)

- Advantages over regular Huffman code
  - compressions/decompression is faster
    - trees have smaller heights
  - compression ratio degrades only a little bit
  - direct searching on a compressed text

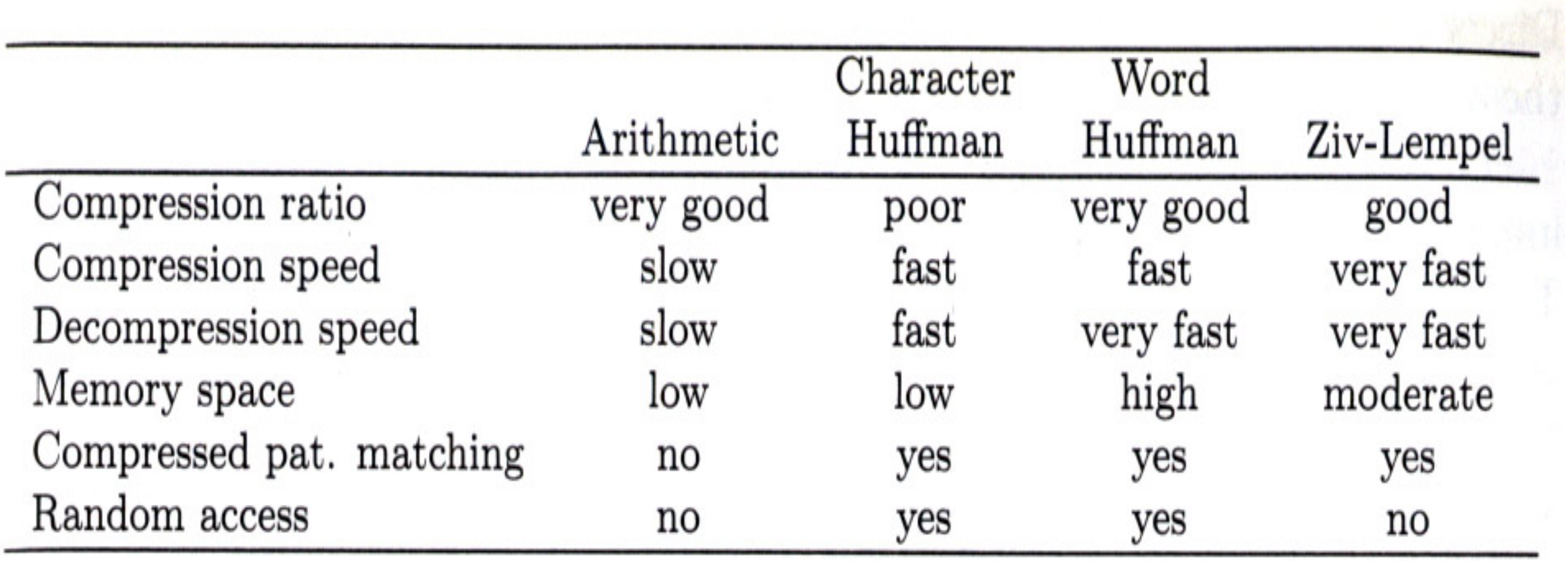
Dictionary Methods

- Replace groups of consecutive symbols (or phrases) with a pointer to an entry in a dictionary
- Approaches
  - static, semi-adaptive, adaptive
- static dictionary
  - simplest, fast
  - e.g., digram coding: selected pairs of letters are replaced with codewords
  - disadvantage: dictionary is suitable only for some text

Inverted File Compression

- An inverted file consists of
  - a vector of distinct words
  - for each word, a list of all documents
    - actual document #s in ascending order
- Can compress the lists
  - Sequence of gaps between document numbers
    - Gaps are small for frequent words
    - Gaps are large for infrequent words
  - Various coding schemes for encoding the gaps
    - Unary
    - Elias-γ, Elias-δ
    - Golomb

Comparison



	Arithmetic	Character Huffman	Word Huffman	Ziv-Lempel
Compression ratio	very good	poor	very good	good
Compression speed	slow	fast	fast	very fast
Decompression speed	slow	fast	very fast	very fast
Memory space	low	low	high	moderate
Compressed pat. matching	no	yes	yes	yes
Random access	no	yes	yes	no

Summary

- Learned about central operations needed for pre-processing documents
  - Lexical analysis, elimination of stopwords, stemming, selection of index terms, construction of term categorisation.
- Learned about text compression methods and how to use them
  - Statistical methods
  - Dictionary methods
  - Inverted file compression