Building Corpora from Scratch

European Masters in Language & Speech, Tutorial 8

Pavel Rychlý

Faculty of Informatics
Masaryk University
Brno, Czech Republic

13–14 July, 2005

Outline of Part I

Outline of Part I

1. Introduction to Text Corpora
2. Using Corpora
   - Lexicography
   - Language Learning
   - Language Modelling
   - Training & Testing & Evaluation of NLP Systems

Outline of Part I

1. Introduction to Text Corpora
2. Using Corpora
   - Lexicography
   - Language Learning
   - Language Modelling
   - Training & Testing & Evaluation of NLP Systems
3. Creating Own Text Corpus
   - Text Selection
   - Corpus Builder

Outline of Part II

4 Textutils/coreutils
- Unix Text Tools
- Text Tools Documentation
- Text Tools Examples
- XML Processing

Outline of Part II

4 Textutils/coreutils
- Unix Text Tools
- Text Tools Documentation
- Text Tools Examples
- XML Processing

5 Regular Expressions

Outline of Part III

Outline of Part III

Outline

What is Text Corpus

purpose Source of language usage examples.

What is Text Corpus

purpose Source of language usage examples.

form

- big collection of texts
- in electronic form
- unified format
- structured
- annotated
- balanced

Corpus Formats

collection/archive	different formats, format depends on text source/type
bank	unified format, document structure, meta-information
vertical text	simple text format with tokenization, one token per line
binary data	used in applications (indexes, statistics)

Character Encoding

8 bit

- 256 characters
- ASCII – 7 bit standard (the base for most 8 bit)
- ISO-Latin standards:
  Western (ISO-8859-1/15),
  Central European (ISO-8859-2), . . .

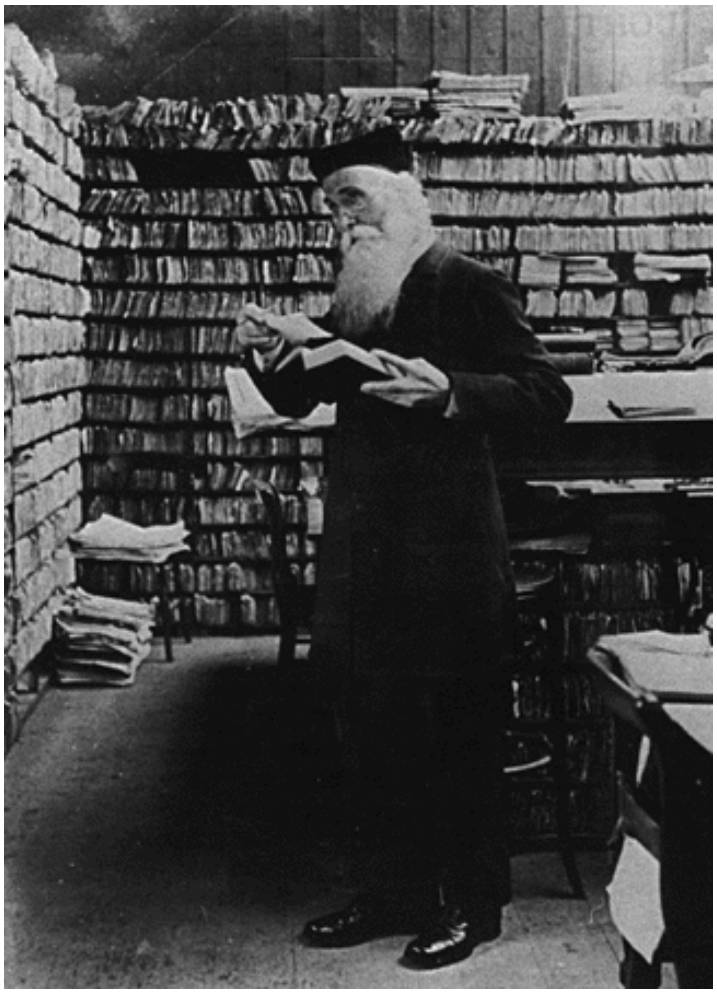
Unicode

- 32 bit per character
- UTF-8 – from 1 to 4 bytes per character

Outline

Pre-computer (Age 1)

Adapted from Adam Kilgarriff's presentation



- Oxford English Dictionary
- 20 million index cards

Corpus Concordancing (Age 2)

KWIC Concordance

```
1 arity, which will be used to take a party  of under-privileged children to D
2  from outside. You are invited to a party and after a couple of drinks you d
3 tion, we believe politicians of all parties will listen to our views.  &equo
4 ould be reaching agreement with all parties  concerned,  as to which events,
5 lack people.  I have certainly been party to one or two discussions amongst
6 . These should be discussed by both parties before entering into the relatio
7 presents They had hosted a cocktail party at Kensington palace,  for example
8 akes. By midnight the end-of-course party  is in full swing,  but most cadet
9 e should be a right for the injured party to terminate the contract.  A mana
10  by the Safran Peoples ' Liberation Party.  This presents the powerful neigh
11 s. Ahead I could see the rest of my party plodding towards the final slope t
12 cial ethic.  The two main political parties - the Tories and the Liberals -
13 ritish successes in Perth The small party of British players competing in th
14 to help control.  One member of the party went to summon the rescue team and
15 rket society fashion magazine.  The party was held at his flat which was a l
16  security and secrecy than any Tory Party Conference : it seems that bootleg
```

- From 1980
- Computerised
- COBUILD project was innovator

▸ try online

Corpus Concordancing (Age 2)

Coloured-Pens Method

arity, which will be used to take a party of under-privileged children to D
from outside. You are invited to a party and after a couple of drinks you d
tion, we believe politicians of all parties will listen to our views. &equo
ould be reaching agreement with all parties concerned, as to which events,
lack people. I have certainly been party to one or two discussions amongst
. These should be discussed by both parties before entering into the relatio
presents They had hosted a cocktail party at Kensington palace, for example
akes. By midnight the end-of-course party is in full swing, but most cadet
e should be a right for the injured party to terminate the contract. A mana
by the Safran Peoples ' Liberation Party. This presents the powerful neigh
s. Ahead I could see the rest of my party plodding towards the final slope t
cial ethic. The two main political parties - the Tories and the Liberals -
ritish successes in Perth The small party of British players competing in th
to help control. One member of the party went to summon the rescue team and
rket society fashion magazine. The party was held at his flat which was a l
security and secrecy than any Tory Party Conference : it seems that bootleg

1 political association
2 social event
3 group of people
4 person in an agreement/dispute
5 to **be party to** something...

Age 2: limitations

As corpora get bigger: too much data

Age 2: limitations

As corpora get bigger: too much data

- 50 lines for a word: read all
- 500 lines: could read all, takes a long time
- 5000 lines: no

Collocations (Age 3)

- Solution:
  list of words occurring in neighbourhood of headword, with frequencies
  - try online

Collocations (Age 3)

- Solution:
  list of words occurring in neighbourhood of headword, with frequencies
  ▸ try online
- Problem:
  too much data - how to summarise?

Collocations (Age 3)

- Solution:
  list of words occurring in neighbourhood of headword, with frequencies
  ▸ try online
- Problem:
  too much data - how to summarise?
- Sorted by salience ▸ try online

Collocations (Age 3)

- Which words?:
  - next word
  - last word
  - window, +1 to +5
  - window, -5 to -1
- How sorted?
- most common collocates –but for most nouns it's the
- most salient collocates –how to measure salience?

Mutual Information

- Church and Hanks 1989
- How much more often does a word pair occur, than one might expect by chance: MI

▸ try online

Mutual Information

- Church and Hanks 1989
- How much more often does a word pair occur, than one might expect by chance: MI

  ▸ try online
- Adjust to emphasise higher-frequency collocates: $MI \times log(jointfrequency)$

Mutual Information

- Church and Hanks 1989
- How much more often does a word pair occur, than one might expect by chance: MI

  ▸ try online
- Adjust to emphasise higher-frequency collocates: $MI \times log(jointfrequency)$
- more measures at *www.collocations.de*

Word Sketch (Age 4)

A corpus-derived one-page summary of a word's grammatical and collocational behaviour ▸ try online

Word Sketch

How to create one

- Large well-balanced corpus
- Parse to find subjects, objects, heads, modifiers etc
- One list for each grammatical relation
- Statistics to sort each list, as before

The Word Sketch Engine

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system
  - Supports complex searching, sorting etc
  - IMS-Stuttgart formalism (also for corpus input)
  - Corpus searches and grammar writing

The Word Sketch Engine Functions

- KWIC concordance
- Sorting, filtering etc
- Word sketch
- Automatic thesaurus
- Sketch difference
  discriminate near-synonyms

Outline

Learning a Foreign Language

- Global world with many languages
- Need to communicate
  - read, write, speak
  - language consumption/production

Tools for Language Learning

- Text books
- Using the language: going abroad
- Dictionaries

Tools for Language Learning

- Text books
- Using the language: going abroad
- Dictionaries
- Good for speaking, reading

Tools for Language Learning

- Dictionary
  - condense knowledge about words
  - limited space
  - only selected features, phrases, examples
- Not enough information
- Collocations (powerful/strong tea)
- Prepositions

Tools for Language Learning

- Dictionary
  - condense knowledge about words
  - limited space
  - only selected features, phrases, examples
- Not enough information
- Collocations (powerful/strong tea)
- Prepositions
- Use Corpus
  - Source of real usage of the language
  - Seach for specific features of words

Outline

Huge area of Language Modelling

- PoS Tagging
- Speech to Text Transcription

Huge area of Language Modelling

- PoS Tagging
- Speech to Text Transcription
- Global statistics of token (word) sequences
- Probability of the following token(s)

Outline

Training & Testing & Evaluation of NLP Systems

- Evaluation (comparison) of NLP systems' performance
- Testing hypothesis, performance, precision, recall, . . .
- Training machine learning tools, . . .

Outline

Text Selection

- Browse web
- Select your papers/books
- Save as plain text

Outline

Corpus Builder

- http://corpora.fi.muni.cz/buildcorp/

Corpus Builder

- http://corpora.fi.muni.cz/buildcorp/
- login/pasword: your last name
- select the first corpus (without *2* suffix)
- upload files
- tag, lematize
- setup web
- test it: try to find words

Outline

Unix Text Tools Tradition

- Unix has tools for text processing from the very beginning (1970s)
- Small, simple tools, each tool doing only one operation
- Pipe (pipeline): powerful mechanism to combine tools

Short Description of Basic Text Tools

cat	concatenate files and print on the standard output
head	output the first part (few lines) of files
tail	output the last part (few lines) of files
sort	sort lines of text files
uniq	remove duplicate lines from a sorted file
comm	compare two sorted files line by line
wc	print the number of newlines, words, and bytes in files
cut	remove sections (columns) from each line of files
join	join lines of two files on a common field
paste	merge lines of files
tr	translate or delete characters

Short Description of Basic Text Tools

egrep	prints lines matching a pattern
(g)awk	pattern scanning and processing language
sed	stream editor, use for substring replacement use `perl -p` for extended regular expressions

Outline

Text Tools Documentation

info run `info` and select from a menu or run directly:

- info coreutils
- info head, info sort, ...
- info gawk

man

- man 7 regex
- man grep, man awk, man tail, ...

–help most tools display a short help message on the `--help` option

- sort --help, uniq --help, ...

Unix Text Tools Packages

Where to find it

- set of system tools
- different sets and different features/options on each Unix type
- GNU textutils
- GNU coreutils – textutils + shellutils + fileutils
- other GNU packages: grep, sed, gawk

Unix Text Tools Packages

Where to find it

- set of system tools
- different sets and different features/options on each Unix type
- GNU textutils
- GNU coreutils – textutils + shellutils + fileutils
- other GNU packages: grep, sed, gawk
- installed on all Linux machines
- on Windows: install mingw32/cygwin, then coreutils, grep, ...

Outline

Text Tools Usage

- command line tools – enter command in a terminal (console) window
- command name followed by options and arguments
- options start with -
- quote spaces and metacharacters: ', ", $
- redirect input and output from/to files using <, >
- use | `less` to only display a result without saving

Text Tools Example 1

task	Convert plain text file to a vertical text.
input	plain.txt
output	plain.vert
solutions	

Text Tools Example 1

task	Convert plain text file to a vertical text.
input	plain.txt
output	plain.vert
solutions	

```
tr -s ' ' '\n' <plain.txt >plain.vert
```

Text Tools Example 1

task Convert plain text file to a vertical text.
input plain.txt
output plain.vert
solutions

```
tr -s ' ' '\n' <plain.txt >plain.vert
```

```
tr -sc a-zA-Z0-9 '\n' <plain.txt >plain.vert
```

Text Tools Example 1

task	Convert plain text file to a vertical text.
input	plain.txt
output	plain.vert
solutions	

```
tr -s ' ' '\n' <plain.txt >plain.vert
```

```
tr -sc a-zA-Z0-9 '\n' <plain.txt >plain.vert
```

```
perl -ne 'print "$&\n" while /(\w+|[^\w\s]+)/g' \
                    plain.txt >plain.vert
```

Text Tools Example 2

task	Create a word list
input	vertical text
output	list of all unique words with frequencies
solutions	

Text Tools Example 2

task Create a word list

input vertical text

output list of all unique words with frequencies

solutions

```
sort plain.vert | uniq -c >dict
sort plain.vert | uniq -c | sort -rn | head -10
```

Text Tools Example 3

task	Corpus/list size
input	vertical text/word list
output	number of tokens/different words
solutions	

Text Tools Example 3

task Corpus/list size

input vertical text/word list

output number of tokens/different words

solutions

```
wc -l plain.vert
wc -l dict
grep -c -i '^[a-z0-9]*$' plain.vert
```

Text Tools Example 4

task	Create a list of bigrams
input	vertical text
output	list of bigrams
solution	

Text Tools Example 4

task Create a list of bigrams

input vertical text

output list of bigrams

solution

```
tail +2 plain.vert |paste - plain.vert \
                   |sort |uniq -c >bigram
```

Text Tools Example 5

task	Filtering
input	word list
output	selected values from word list
solutions	

Text Tools Example 5

task Filtering

input word list

output selected values from word list

solutions

```
grep '^[0-9]*$' dict
awk '$1 > 100' dict
```

Text Tools Debuging

- data driven programming
- cut the pipline a display partial results
- try single command with a test input

Text Tools Exercise

task Find all words from a word list differing with s/z alternation only: apologize/apologise

Text Tools Exercise

task Find all words from a word list differing with s/z alternation only: apologize/apologise

solutions

```
tr s z < dict | sort |uniq -d >szaltern
```

Text Tools Exercises

- Find all words from a word list differing with
  s/z alternation only,
  and each alternation has higher frequency than 50

Text Tools Exercises

- Find all words from a word list differing with s/z alternation only, and each alternation has higher frequency than 50
- and display their frequences

Text Tools Exercises

- Find all words from a word list differing with s/z alternation only, and each alternation has higher frequency than 50
- and display their frequences
- Find all words which occurs in the word list only with capital letter (names).

Outline

XML Processing

- XML is text format, use text tools
- API

  SAX Simple API for XML
  DOM Document Object Model

XML API SAX

Simple API for XML

- event driven processing
- events:
  - start/end of an element
  - element attribute (with value)
  - text
- calls a function/method for each event
- minimal memory requirements, suitable for large documents

XML API DOM
Document Object Model

- XML document stored as a tree
- methods for accessing (finding/traversing) document parts
- tree modification methods
- whole structure in memory
- very good for random access

Regular Expression Basics

- RE – pattern that describes a set of strings
- most characters matches itself
- meta-characters – special meaning

.	The period '.' matches any single character.
?	The preceding item is optional and will be matched at most once.
*	The preceding item will be matched zero or more times.
[and]	Character classes – matches any single character in the list.
^ and $	Matches the empty string at the beginning/end of a line or string.

Regular Expression Documentation

- read documentation
- info grep
- man 7 regex

Outline

Part of Speech Tagging

- adding more information to corpus
- getting much better results
  - local structure, finding specific features
  - global structure, more attributes to model

Part of Speech Tagging

Tagger Types

- statistical
- rules based

Part of Speech Tagging
Tagger Types

- statistical
- rules based
- Brill's tagger
  - very good if trained on a small corpus

Part of Speech Tagging
Tagger Types

- statistical
- rules based
- Brill's tagger
  - very good if trained on a small corpus
- combinations

Tag-set

- if there is a tagger, use it
- think about future purpose/applications
- simple tag-set is better
- complex tag-set can be reduced

Outline

Lemmatization

- usage depends on language

Lemmatization

- usage depends on language
- many languages don’t need it:
  Chinese, English (use case folding)

Lemmatization

- usage depends on language
- many languages don’t need it: Chinese, English (use case folding)
- for many languages it is a necessity: Czech

Lemmatizers

- many taggers provide lemmatization

Lemmatizers

- many taggers provide lemmatization
- from PoS tagged corpus:
  could be a set of regular expression substitutions

Question

Do you have a tagger and lemmatizer for your language?

Outline

The Word Sketch Engine

Summary from the first part

- Input:
  - any corpus, any language
  - Lemmatised, part-of-speech tagged
  - specification of grammatical relations
- Word sketches integrated with
- Corpus query system
  - Supports complex searching, sorting etc
  - IMS-Stuttgart formalism (also for corpus input)
  - Corpus searches and grammar writing

Corpus Query Language

- Query – pattern matching a set of single tokens or token sequences

Corpus Query Language

- Query – pattern matching a set of single tokens or token sequences
- Each token consists of attributes (depending on corpus configuration).
- Use *[attribute=”value”]* for each token sub-pattern.

CQL Examples 1

- Test examples at http://corpora.fi.muni.cz/bnc/ (login/password: emasters)
- *New query* link or *Concordance* button
- CQL entry box

```
[word="dream"]
[word="Dream"]
[lc="dream"]
[lemma="dream"]
[lempos="dream-n"]
[word="The"] [word="dream"]
[word="the"] [lemma="dream"]
[tag="AJ0"] [lempos="dream-n"]
```

CQL Examples 2

Value is a regular expression in a *[attribute="value"]* expression.

```
[word="dream.*"]
[word="[dD]ream"]
[word="[0-9]*"] [lc="dreams"]
[tag="NN."] [lempos="dream-v"]
[word="[0-9]{5,}"] [word="\."]
[word="\("] [word="0[0-9]{3}"] [word="\)"]

[word="[A-Z][0-9A-Z]{2,3}"] [word="[0-9][0-9A-Z]{2}"]
```

CQL Examples 3

Boolean combinations (*AND*, *OR* and *NOT*) of *[attribute="value"]* expressions.
Use: &, |, !=, ()

```
[word="dream" & tag="NN1"]
[lemma="dream" & tag="VV."]
[word="dream" | word="Dream"]

[word="the" | tag="DPS"][lempos="dream-n" & tag="NN2"]
[word="the" | (tag="DPS" & lemma!="my")][lemma="dream"]
```

CQL Examples 4

Regular expressions on token level:

- `?` optional token
- `*` any number of repetition
- `{N}` exact number of repetition
- `[]` any token

```
[tag="DPS"] [] [lemma="dream"]
[tag="DPS"] [tag="AJ0"]? [lemma="dream"]
[tag="AJ0"]{2} [lemma="dream"]
[word="the"] []{0,3} [lempos="dream-n"]
```

CQL Examples 5

within keyword at the end of a query

- `within <s>` restricts result to one sentence
- `within <bncdoc id="A0.">` restricts result to a subcorpus

```
[lemma="dream"] within <bncdoc id="A0.">
[word="the"] []{3,5} [lemma="dream"]
[word="the"] []{3,5} [lemma="dream"] within <s>
```

CQL Examples 6

More *within* combinations

```
[lemma="dream"] within <bncdoc author=".*Smith.*">

[lemma="dream"] within <bncdoc wriaud="Teenager"
                              & wriase="Female">

[word="the"] []{3,5} [lemma="dream"]
                within <s> within <bncdoc id="A0.">
```

CQL Examples 7

Structure boundaries

```
<s> [lemma="dream"]
[word="\?"] </bncdoc>
<head /> within <bncdoc alltyp="Written-to-be-spoken">
```

CQL Examples 8

Global condition

- numeric labels of tokens
- testing agreement or disagreement of attribute values

```
[tag!="NN."] [word="and"] [tag!="NN."]
```

CQL Examples 8

Global condition

- numeric labels of tokens
- testing agreement or disagreement of attribute values

```
[tag!="NN."] [word="and"] [tag!="NN."]
```

```
1:[tag!="NN."] [word="and"] 2:[tag!="NN."]
                          & 1.tag = 2.tag
```

Outline

Grammatical Relations Definition

- plain text file
- a set of queries for each GR
- queries contain labels for keyword and collocate
- processing options

GR Definition Examples

```
# ‘adverb’ gramrel definition
=adverb
   1:[] 2:"AV."
   2:"AV." 1:[]

# ‘and/or’ gramrel definition
=and/or
*SYMMETRIC
   1:[] [word="and"|word="or"] 2:[] & 1.tag = 2.tag
```

GR Definition Examples

```
# ‘modifier’ and ‘modify’ gramrels definition
*DUAL
=modifier/modify
   2:"AJ." 1:"N.."

*UNARY
=wh_word
1:[] [tag="AVQ"|tag="DTQ"|tag="PNQ"]

*TRINARY
=pp_%s
1:[tag="N.."|tag="AJ."] 3:"PR." 2:"N.."
```

Summary

Summary

- Use simple Unix text tools for processing text files and computation of global statistics.

Summary

- Use simple Unix text tools for processing text files and computation of global statistics.
- Use a powerful graphical user interface for local corpus exploration:

Summary

- Use simple Unix text tools for processing text files and computation of global statistics.
- Use a powerful graphical user interface for local corpus exploration:
  - Word Sketch Engine: www.sketchengine.co.uk

Summary

- Use simple Unix text tools for processing text files and computation of global statistics.
- Use a powerful graphical user interface for local corpus exploration:
  - Word Sketch Engine: www.sketchengine.co.uk
  - Manatee/Bonito: www.textforge.cz