Part II — Semistructured Data

XML:

Corpora:

Applications of corpora

Answering *empirical questions* in linguistics and cognitive science:

- corpora can be analyzed using statistical tools;
- hypotheses about language processing and language acquisition can be tested;
- new facts about language structure can be discovered.

Engineering *natural-language systems* in AI and computer science:

- corpora represent the data that these language processing systems have to handle;
- algorithms can find and extract regularities from corpus data;
- text-based or speech-based computer applications can learn automatically from corpus data.

Extracting data from corpora

To do something useful with corpus data and its annotation, we need to be able to query the corpus to extract the data and information we want.

This lecture introduces:

- The basic notion of a *concordance* in a corpus.
- Statistics of *frequency* and *relative frequency*, useful for linguistic questions and natural language processing.
- *Unigrams*, *bigrams* and *n-grams*.
- Applications of corpora in informatics
- The linguistic notion of a *collocation*.

Concordances

Concordance: all occurrences of a given word, displayed in context.

More generally, one looks for all occurrences of matches for some query expression.

- generated by concordance programs based on a user keyword;
- keyword (search query) can specify word, annotation (POS, etc.) or more complex information (e.g., using regular expressions);
- output displayed as keyword in context: matched keyword in the middle of the line, with a fixed amount of context to left and right.

Example

A concordance for all forms of the word *"remember"* in a corpus of the complete works of Dickens.

```
's cellar .  Scrooge then <remembered> to have heard that ghost
, for your own sake , you <remember> what has passed between
e-quarters more , when he <remembered> , on a sudden , that the
corroborated everything , <remembered> everything , enjoyed eve
urned from them , that he <remembered> the Ghost , and became c
ht be pleasant to them to <remember> upon Christmas Day , who
its festivities ; and had <remembered> those he cared for at a
wn that they delighted to <remember> him .  It was a great sur
ke ceased to vibrate , he <remembered> the prediction of old Ja
as present myself , and I <remember> to have felt quite uncom
...
```

Example

A concordance for all occurrences of *"Holmes"* in a corpus that consists of the Arthur Conan Doyle story *A Case of Identity*.

```
My dear fellow." said Sherlock <Holmes> as we sat on either
a realistic effect," remarked <Holmes>.  "This is wanting in the
said <Holmes>, taking the paper and glancing his eye down
"I have seen those symptoms before," said <Holmes>, throwing
merchant-man behind a tiny pilot boat.  Sherlock <Holmes> welcomed
You've heard about me, Mr.  <Holmes>," she cried, "else how
...
```

Frequencies

Frequency information obtained from corpora can be used to investigate characteristics of the language represented.

Token count N: number of tokens (words, punctuation marks, etc.) in a corpus (i.e., size of the corpus).

Type count: number of *different* tokens in a corpus.

Absolute frequency $f(t)$ *of a type* t: number of tokens of type t in a corpus.

Relative frequency of a type t: absolute frequency of t normalized by the token count, i.e., $f(t)/N$.

Here a *type* might be a single word, or its variants, or a particular part of speech.

Frequencies (example)

The British National Corpus (BNC) is an important reference.

Let's compare some counts from the BNC with counts from our sample corpus *A Case of Identity*

	BNC	A Case of Identity
Token count N	100,000,000	7,006
Type count	636,397	1,621
f(Holmes)	890	46
f(Sherlock)	209	7
f(Holmes)$/N$	.0000089	.0066
f(Sherlock)$/N$	.00000209	.000999

Unigrams

We can now ask questions such as: what are the most frequent words in a corpus?

- Count absolute frequencies of all word types in the corpus;
- tabulate them in an ordered list;
- results: list of *unigram* frequencies (frequencies of individual words).

The next slide compares unigram frequencies for BNC and *A Case of Identity*.

Unigrams (example)

BNC		A Case of Identity	
6,184,914	the	350	the
3,997,762	be	212	and
2,941,372	of	189	to
2,125,397	a	167	of
1,812,161	in	163	a
1,372,253	have	158	I
1,088,577	it	132	that
917,292	to	117	it

N.B. The article "the" is the most frequent word in both corpora; prepositions like "of" and "to" appear in both lists; etc.

n-grams

The notion of unigram can be generalized:

- *bigrams* — pairs of adjacent words
- *trigrams* — triples of adjacent words
- n*-grams* — n-tuples of adjacent words.

As the value of n increases, the units become more linguistically meaningful.

n-grams (example)

Compute the most frequent n-grams in *A Case of Identity*, for $n = 2, 3, 4$.

bigrams		trigrams		4-grams	
40	of the	5	there was no	2	very morning of the
23	in the	5	Mr. Hosmer Angel	2	use of the money
21	to the	4	to say that	2	the very morning of
21	that I	4	that it was	2	the use of the
20	at the	4	that it is	2	the King of Bohemia

N.B. n-gram frequencies get smaller with increasing n. As more word combinations become possible, there is increased *data sparseness*.

Example

A concordance for all occurrences of bigrams in the Dickens corpus in which the second word is *"tea"* and the first is an adjective.

This query exploits the POS tagging of the corpus to search for adjectives.

```
now , notwithstanding the <hot tea> they had given me before
." Shall I put a little <more tea> in the pot afore I go ,
o moisten a box-full with <cold tea> , stir it up on a piece
tween eating , drinking , <hot tea> , devilled grill , muffi
e , handed round a little <stronger tea> .  The harp was there ; t
e so repentant over their <early tea> , at home , that by eigh
rs.  Sparsit took a little <more tea> ; and , as she bent her
s illness !  Dry toast and <warm tea> offered him every night
of robing , after which , <strong tea> and brandy were administ
rsty .  You may give him a <little tea> , ma'am , and some dry t
```

Applications: Corpora in Informatics

Corpora are used extensively in two areas of informatics:

- *Natural Language Processing (NLP)* builds computer systems that understand or produce text. Example applications that rely on corpus data include:
  - *Summarization:* take a text and compress it, i.e., produce an abstract or summary. Example: Newsblaster.
  - *Machine Translation (MT):* take a text in a source language and turn it into a text in the target language. Example: Babel Fish.
- *Speech Processing* systems that understand or produce spoken language.

The techniques applied rely on probability theory, information theory and machine learning to extract statistical regularities from corpora.

Featured application: machine translation

Machine translation maps a *source sentence* in one language (called the *source language*) to a corresponding *target sentence* in another language (called the *target language*). The aim is to preserve meaning.

Two major approaches:

1. *Rule-based translation*

   This is the approach used by Babel Fish

2. *Statistical translation*

   This is the approach used by Google translate

Both approaches make use of corpora.

Rule-based machine translation

1. Automatically assign part-of-speech information to source sentence.
2. Parse source sentence (that is, build its syntax tree), using a collection of grammatical rules.
3. Map parse tree in source language to translated sentence, using a dictionary to perform translation at the word level, and a collection of rules to infer correct inflections and word ordering for translated sentence.

Corpora are used, in the development of rule-based translation systems, as a source for machine learning of grammatical structures.

Example rule-based translation by Yahoo! Babel Fish

O, my love is like a red, red rose,
That's newly sprung in June.

Robert Burns (1759–1796)

English ⟶ Italian:

La O, il mio amore 'e come un rosso, colore rosso 'e aumentato,
That's recentemente balzata in giugno.

Italian ⟶ English:

Or, my love is like a red one, red color is increased,
That's recently jumped in june.

Issues with rule-based translation

A major difficulty with rule-based translation is to include a large enough collection of rules to sufficiently cover the very many special cases and nuances in natural language usage.

As a result of this, rule-based translations often have a very unnatural feel.

This isse is a major one, and rule-based translation systems have not yet overcome this problem.

(The problem with the example translation on the previous page is of a different nature. The source text is poetry, and rule-based translation is not designed for poems, where huge liberties are taken with grammar and use of vocabulary.)

Statistical machine translation

Uses a corpus of *parallel texts* (the same text rendered in both source and target language).

1. Match words and phrases from the source sentence with occurrences of the word or phrase in the corpus.
2. Look at the translations of the matched words and phrases, as they occur in the parallel texts, and use statistical methods to infer preferred (most likely) translations.
3. Some *smoothing* is done to find appropriate unit sizes for phrases and to glue translated phrases together to produce the translated sentence.

To be effective, statistical translation requires a large and representative corpus of parallel texts. This corpus does not need to be annotated.

Example statistical translation with Google translate

O, my love is like a red, red rose,
That's newly sprung in June.

Robert Burns (1759–1796)

English ⟶ Italian:

Oh, mio amore come un rosso, rosa rossa,
Quello appena nata nel mese di giugno.

Italian ⟶ English:

Oh, my love is like a red, red rose,
That's just born in June.

Issues with statistical translation

Difficulties with statistical translation: it requires a very large corpus of parallel texts, and is computationally expensive to carry out. In recent years, these problems have diminished: large corpora have become available, and improvements to algorithms and hardware have helped with efficiency,

Given a large enough corpus, statistical translations tend to produce more natural translations than rule-based translations.

Because it is not tied to grammar, statistical translation can work better with less rigid uses of language (such as poetry).

However, if statistical translation is applied to a sentence that uses uncommon phrases, not in the corpus, then it can result in nonsense, whereas machine-based translation is likely to be more effective.

Currently, statistical translation is the preferred technology.

Featured linguistic application: finding collocations

Collocation: a sequence of words that occurs 'atypically often' in language usage

Examples:

- *run amok:* the verb "run" can occur on its own, but "amok" can't.
- *strong tea:* sounds much better than "powerful tea" although the literal meanings are much the same.
- Phrasal verbs such as *make up* or *make off* or *make out* (but not, for example, "make in").
- *rancid butter*, *bitter sweet*, *over and above*, etc.

N.B. The inverted commas around 'atypically often' are because we need statistical ideas to make this precise.

Identifying collocations

Task: automatically identify collocations in a large corpus.

For example collocations with the word *tea* (see III: 109).

- *strong tea* occurs in the corpus.

  This is a collocation.

- *powerful tea*, in fact, does not.

- However, *more tea* and *little tea* also occur in the corpus.

  These are not collocations. These word sequences do not occur with an *atypically* common frequency.

Problem: How do we detect when a bigram (or n-gram) is a collocation?

Looking at the data

The next slide lists the frequencies of the most common bigrams, in the Dickens Corpus, in which the first word is *"strong"*.

For comparison, the frequencies of the most common bigrams in which the first word is *"powerful"* are also given.

strong	and	31	powerful	effect	3
	enough	16		sight	3
	in	15		enough	3
	man	14		mind	3
	emphasis	11		for	3
	desire	10		and	3
	upon	10		with	3
	interest	8		enchanter	2
	a	8		displeasure	2
	as	8		motives	2
	inclination	7		impulse	2
	tide	7		struggle	2
	beer	7		grasp	2

Filtering collocations

The bigram table shows:

- Neither *strong tea* nor *powerful tea* are frequent enough to make it into the top 13.
- Potential collocations for *strong*: e.g., *strong desire*, *strong inclination*, and *strong beer*;
- Potential collocations for *powerful*: e.g., *powerful effect*, *powerful motives*, and *powerful struggle*;
- Problem: The bigrams *strong and*, *strong enough*, *powerful for*, are highly frequent. These are not collocations.
- To distinguish collocations from non-collocations, we need to filter out 'noise'.

The need for statistics

Problem: Words like *for* and *and* are highly frequent on their own: they occur with *tea* by chance.

Solution: use statistical testing to detect when the frequency of a bigram is *atypically high* given the frequencies of its constituent words.

In general, statistical tools offer powerful methods for the analysis of all types of data. In particular, they provide the principal approach to the quantitative (and qualitative) analysis of *unstructured data*.

We shall return to the problem of finding collocations in Part III of the course, when we have appropriate statistical tools at our disposal.

Searching for concordances

The concordances in this lecture were produced using a dedicated program for searching for concordances, the *Corpus Query Processor (CQP)*.

CQP is query engine which searches corpora based on user queries over words, parts of speech, or other markup.

It uses *regular expressions* to formulate queries. This makes the CQP query language very powerful

An alternative to using a dedicated concordance program is to use XML query technology (XPath and XQuery) to search any corpus implemented in XML.