Informatics 1: Data & Analysis

Lecture 13: Annotation of Corpora

Ian Stark

School of Informatics
The University of Edinburgh

Friday 8 March 2013
Semester 2 Week 7

Lecture Plan

XML

We start with technologies for modelling and querying *semistructured data*.

- Semistructured Data: Trees and XML
- Schemas for structuring XML
- Navigating and querying XML with XPath

Corpora

One particular kind of semistructured data is large bodies of written or spoken text: each one a *corpus*, plural *corpora*.

- Corpora: What they are and how to build them
- Applications: corpus analysis and data extraction

Homework

Coursework Assignment

The coursework assignment is now online. This runs alongside your usual tutorial exercises for the next two weeks; ask tutors for help with it where you have specific questions.

The assignment is a Inf1-DA examination paper from 2011. Your tutor will give you marks and feedback on your work in the last tutorial of semester.

Reading

T. McEnery and A. Wilson.
Corpus Linguistics. Second edition, Edinburgh University Press, 2001.
Chapter 2: What is a corpus and what is in it? (§2.2.2 optional)

Still available from the ITO.

Corpus Annotation

Last lecture introduced the *preprocessing* steps of identifying tokens and sentence boundaries. Now we look to add further information to the data.

Annotation adds information to the corpus that is not explicit in the data itself. This is often specific to a particular application; and a single corpus may be annotated in multiple ways.

Annotation scheme is a basis for annotation, made up of a *tag set* and *annotation guidelines*.

Tag set is an inventory of labels for markup.

Annotation guidelines tell annotators — domain experts — how a tag set should be applied. In particular, this is to ensure consistency across different annotators.

Part-of-Speech (POS) Annotation

Tagging by *part-of-speech* (POS) is the most basic kind of linguistic annotation.

Each token is assigned a code indicating its part of speech. This might be a very simple classification:

- Noun ("claw", "hyphen");
- Adjective ("red", "small");
- Verb ("encourage", "betray").

Or it could be more refined:

- Singular common noun ("elephant", "table");
- Comparative adjective ("larger", "neater");
- Past participle ("listened", "written").

Even simple POS tagging can, for example, disambiguate some *homographs* like "boot" (verb) and "boot" (noun).

Example POS Tag Sets

- CLAWS tag set (used for BNC): 62 tags

  (Constituent Likelihood Automatic Word-tagging System)
- Brown tag set (used for Brown corpus): 87 tags
- Penn tag set (used for the Penn Treebank): 45 tags

Category	Examples	CLAWS5	Brown	Penn
Adjective	happy, bad	AJ0	JJ	JJ
Adverb	often, badly	PNI	CD	CD
Determiner	this, each	DT0	DT	DT
Noun	aircraft, data	NN0	NN	NN
Noun singular	goose, book	NN1	NN	NN
Noun plural	geese, books	NN2	NN	NN
Noun proper singular	London, Michael	NP0	NP	NNP
Noun proper plural	Greeks, Methodists	NP0	NPS	NNPS

POS Tagging

Idea: Tag parts of speech by looking up words in a dictionary.

Problem: Ambiguity: words can carry several possible POS.

Time flies like an arrow (1) / Fruit flies like a banana (2)

time: singular noun or a verb;
flies: plural noun or a verb;
like: singular noun, verb, preposition.

Combinatorial explosion: $2 \times 2 \times 3 = 12$ POS sequences for (1).

To resolve this kind of ambiguity, we need more information. One route would be to investigate the meaning of words and sentences — their *semantics*.

Perhaps unexpectedly, it turns out that impressive improvements are possible using only the *probabilities* of different parts of speech.

Probabilistic POS Tagging

Observation: words can have more than one POS, but one may be more frequent than the others.

Idea: assign each word its most frequent POS (using frequencies from manually annotated training data). Accuracy: around 90%.

Improvement: use frequencies of POS *sequences*, and other context clues. Accuracy: 96–98%.

Sample POS tagger output

It/PP was/VBD the/DT best/JJS of/IN times/NNS ,/, it/PP was/VBD the/DT worst/JJS of/IN times/NNS ,/, it/PP was/VBD the/DT age/NN of/IN wisdom/NN ,/, it/PP was/VBD the/DT age/NN of/IN foolishness/NN ,/, it/PP was/VBD the/DT epoch/NN of/IN belief/NN ,/, it/PP was/VBD the/DT epoch/NN of/IN incredulity/NN ,/, it/PP was/VBD the/DT season/NN of/IN Light/NP ,/, it/PP was/VBD the/DT season/NN of/IN Darkness/NN

Data and Metadata

One important application of markup languages like XML is to separate *data* from *metadata*:

Data is the thing itself.

In a corpus this is the samples of text.

Metadata is data about the data.

In a corpus this includes information about source of text as well as various kinds of annotation.

At present XML is the most widely used markup language for corpora, replacing various others including the *Standard Generalized Markup Language* (SGML).

The example on the next slide is taken from the BNC, which was first released as XML in 2007 (having been previously formatted in SGML).

Example from BNC XML Edition

```
<wtext type="FICTION">
    <div level="1">
        <head> <s n="1">
            <w c5="NN1" hw="chapter" pos="SUBST">CHAPTER </w>
            <w c5="CRD" hw="1" pos="ADJ">1</w>
        </s> </head>
        <p> <s n="2">
            <c c5="PUQ">'</c>
            <w c5="CJC" hw="but" pos="CONJ">But</w>
            <c c5="PUN">,</c> <c c5="PUQ">'</c>
            <w c5="VVD" hw="say" pos="VERB">said </w>
            <w c5="NP0" hw="owen" pos="SUBST">Owen</w>
            <c c5="PUN">,</c> <c c5="PUQ">'</c>
            <w c5="AVQ" hw="where" pos="ADV">where </w>
            <w c5="VBZ" hw="be" pos="VERB">is </w>
            <w c5="AT0" hw="the" pos="ART">the </w>
            <w c5="NN1" hw="body" pos="SUBST">body</w>
            <c c5="PUN">?</c> <c c5="PUQ">'</c>
        </s> </p>
        ....
    </div>
</wtext>
```

Aspects of BNC Example

This example is the opening words of BNC text J10, which is a novel: *The Mamur Zapt and the girl in the Nile* by Michael Pearce.

- The wtext element stands for *written text*. Its attribute type indicates the kind of text (here FICTION).
- Element head tags a portion of header text (here, a chapter heading).
- The s element tags sentences. Sentences are numbered via the attribute n.
- The w element tags words. The attribute pos is a POS tag, with more detailed POS information given by the c5 attribute, which contains the CLAWS code. Attribute hw represents the *head word*, *lemma* or *root form* of the word (e.g., the root form of "said" is "say").
- The c element tags punctuation.

Syntactic Annotation

Moving above the level of individual words, *parsing* and *syntactic annotation* give information about the structure of sentences.

Linguists use *phrase markers* to indicates which parts of a sentence belong together:

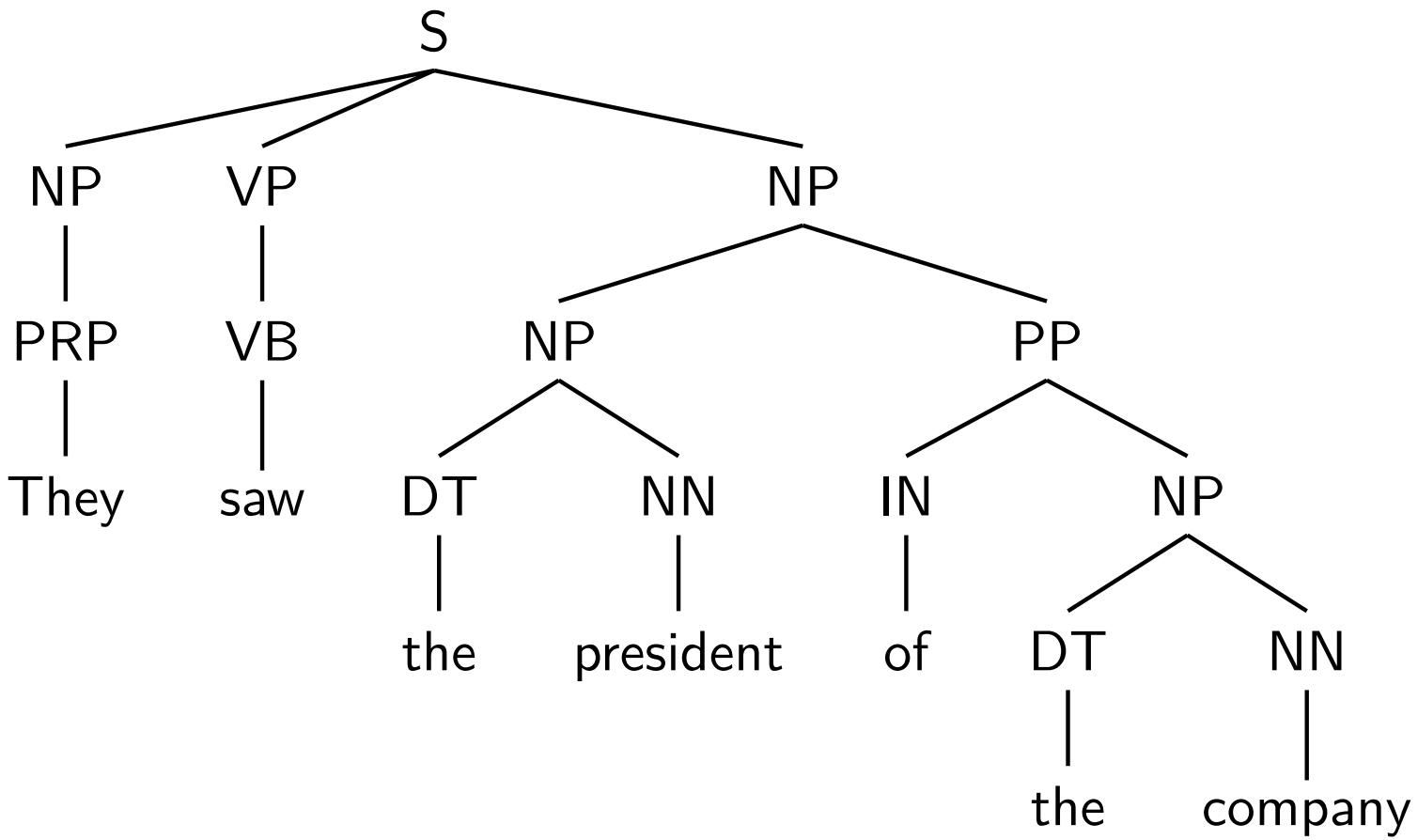
- noun phrase (NP): noun and its adjectives, determiners, etc.
- verb phrase (VP): verb and its objects;
- prepositional phrase (PP): preposition and its NP;
- sentence (S): VP and its subject.

Phrase markers group hierarchically into a *syntax tree*.

Syntactic annotation can be automated. Accuracy: around 90%.

Example Syntax Tree

The following is from the Penn Treebank corpus.



Syntax Tree in XML

Here is the same syntax tree expressed in XML:

```
<s>
    <np><w pos="PRP">They</w></np>
    <vp><w pos="VB">saw</w>
        <np>
            <np><w pos="DT">the</w>
                <w pos="NN">president</w></np>
            <pp><w pos="NN">of</w>
                <np><w pos="DT">the</w>
                    <w pos="NN">company</w></np>
            </pp>
        </np>
    </vp>
</s>
```

Some choices made in this XML coding: phrase markers are represented by XML elements; while POS tags are given by attribute values.

Note that, as a result of this, the tree on the previous slide is *not* quite the same as the XML element tree for this document.

Examinable Material

Unless otherwise specified, all of the following material is examinable:

- Topics covered in lectures
- Directed reading distributed in lectures
- Topics covered in the weely exercise sheets

All coursework, including the weekly exercise sheets, is compulsory. All coursework in Inf1-DA is *formative*: any marks for such coursework are for feedback only, to aid learning, and do not contribute to the final mark for the course. Nevertheless, doing the coursework is essential to gaining a proper understanding of lecture material. If you do not do the coursework then you are very unlikely to pass the exams. Some exam questions may be directly based on previous coursework questions.

Burning Questions

Burning Questions

- Will this be on the exam?

Burning Questions

- Will this be on the exam?
- Is this examinable?

Burning Questions

- Will this be on the exam?
- Is this examinable?
- Do I need to know this?

Burning Questions

- Will this be on the exam?
- Is this examinable?
- Do I need to know this?
- Is it important?

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 Information from Corpora

Once we have an annotated corpus, we can begin to use it to find out information and answer questions. For now, we start with the following:

- The basic notion of a *concordance* in a text.
- Statistics of word *frequency* and *relative frequency*, useful for linguistic questions and natural language processing.
- Word groups: *Unigrams*, *bigrams* and *n-grams*.

The next lecture will look at more substantial examples of detecting *collocations* and the *machine translation* of natural language.

Concordances

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

More generally, a concordance may extend to all matches for some query expression.

- Specialist concordance programs will generate these from a given keyword.
- This might can specify word, annotation (POS, etc.) or more complex information (e.g., using regular expressions).
- Results are typically displayed as keyword in context (kwik): a matched keyword in the middle of a line with a fixed amount of context to left and right.

Example Concordance

This is a concordance for all forms of the word "remember" in the works of Dickens, generated by the *Corpus Query Processor* cqp.

```
'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
...
```

Frequencies

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

- Token count N: the number of tokens (words, punctuation marks, etc.) in a corpus; i.e., the size of the corpus.
- Type count: the number of types of token in a corpus.
- Absolute frequency $\mathrm{f(t)}$ of type t: the number of tokens of type t in a corpus.
- Relative frequency of type t: the absolute frequency of t scaled by the token count, i.e., $\mathrm{f(t)/N}$.

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

Frequency Example

Here is a comparison of frequency information between two sources: the BNC and the Sherlock Holmes story *A Case of Identity* by Sir Arthur Conan Doyle.

	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	0.0000089	0.0066
f("Sherlock")/N	0.00000209	0.000999

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.
- Result: list of *unigram* frequencies — frequencies of individual words.

Unigram 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

The unigram rankings are different, but we can see similarities. For example, the definite article "the" is the most frequent word in both corpora; and prepositions like "of" and "to" appear in both lists.

$\mathfrak{n}$-grams

The notion of unigram generalizes:

- Bigrams — pairs of adjacent words;
- Trigrams — triples of adjacent words;
- $\mathfrak{n}$-grams — $\mathfrak{n}$-tuples of adjacent words.

These larger clusters of words carry more linguistic significance than individual words; and, again, before seeking to understand their semantic content.

n-grams example

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

Note that frequencies of even the most common n-grams naturally get smaller with increasing n. As more word combinations become possible, there is an increase in *data sparseness*.

Bigram and POS Example Concordance

Here is 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 use the POS tagging of the corpus to search for adjectives.

[pos="J.*"][word="tea"]

```
  87773: now , notwithstanding the <hot tea> they had given me before
 281162: .’ ’ Shall I put a little <more tea> in the pot afore I go ,
 565002: o moisten a box-full with <cold tea> , stir it up on a piece
 607297: tween eating , drinking , <hot tea> , devilled grill , muffi
 663703: e , handed round a little <stronger tea> . The harp was there ;
 692255: e so repentant over their <early tea> , at home , that by eigh
1141472: rs. Sparsit took a little <more tea> ; and , as she bent her
1322382: s illness ! Dry toast and <warm tea> offered him every night
1456507: of robing , after which , <strong tea> and brandy were administ
1732571: rsty . You may give him a <little tea> , ma’am , and some dry t
```