Corpus annotation

Beáta B. Megyesi

Uppsala University
Department of Linguistics and Philology
beata.megyesi@lingfil.uu.se

Last time

- Using available corpora
- Corpus representativeness
- Corpus sampling
- Corpus types
- Corpus usage in language studies:
  - concordances
  - collocations
  - frequency lists
  - keywords

Outline

- Corpus mark-up
- Corpus annotation
- Treebanks

Corpus mark-up

- Preparation of raw corpus:
  - collected, sampled text: machine-readable or OCR scanning or keyboarding
  - spoken data transcribed from audio recordings
- Mark-up the corpus to provide information about the text itself as its contextual info is lost
- Filename is not enough as they provide
  - a tiny amount of extratextual info (text types, speakers) and
  - no textual info (paragraph/sentence boundaries, speech turns)

Mark-up schemes

- COCOA (Word COunt and COncordance on Atlas): earliest
  - a set of attribute names and values enclosed in angle brackets
  - `<N LET TO HUSBAND> N (title)`
  - `<A BEAMONT ELIZABETH> A (author)`
  - `<X FEMALE> X (gender)`
  - `<H HIGH> H (social status)`
  - `encode only limited set of features (authors, titles and dates)`
- Most influental schemes:
  - TEI: Text Encoding Initiative
  - CES: Corpus Encoding Standard

TEI

- facilitate data exchange by standardizing the mark-up or encoding
- http://www.tei-c.org.index.html
- each text consists of 2 parts: head and body (the text itself)

Headers

- corpus header
  - information about the corpus project (project management, assistants), project description, annotation, tagset with example, genres)
- file header
  - file description <fileDesc>: full bibiliography of the file
  - encoding description <encodingDesc>: the source from which the text was derived
  - text profile <profileDesc>: language, sublanguages, participants, settings
  - revision history <revisionDesc>: changes that have been made to the file

TEI header example

<titleStmt>
<title level=a>Smygrustning av raketvapen</title>
<title level=j>Dagens Nyheter</title>
<author>Mats Lundegard</author>
</titleStmt>

TEI file example

-
- <s id=aa01a-009>
- <w n=91>Hur<ana><ps>HA<b>hur</w>
- <w n=92>är<ana><ps>VB<m>PRS AKT<b>vara</w>
- <w n=93>det<ana><ps>PN<m>NEU SIN DEF SUB/OBJ<b>det</w>
- <w n=94>da<ana><ps>AB<b>da</w>
- <w n=95>i<ana><ps>PP<b>i</w>
- <name type=place>
- <w n=96>Mellanöstern<ana><ps>PM<m>NOM<b>Mellanöstern</w>
- </name>
- <d n=97>?<ana><ps>MAD<b>?</d>
- </s>
-

CES

- designed specifically for the encoding of language corpora
- simplified TEI; includes a subset of TEI tags necessary for corpus-based work
- mark-up

  document-wide mark-up: bibliographic description, encoding

  structural mark-up: encodes structural units of text (volume, chapter), paragraphs, footnotes, titles, headings, tables, figures

  sub-paragraph structures: sentences, quotations, words, abbreviations, names, dates, terms, cited words

Annotation format: XCES

```
<s id="s7.29">
<w id="w7.29.1">Islam</w>
<w id="w7.29.2">är</w>
<w id="w7.29.3">ingen</w>
<w id="w7.29.4">ny</w>
<w id="w7.29.5">företeelse</w>
<w id="w7.29.9">.</w>
</s>
```

Corpus annotation

- closely related to corpus mark-up
- the process of adding linguistic information to a corpus
- Advantages:
  - extract information easier from annotated corpora (saw / Verb or Noun)
  - reusable resource (annotation is time-consuming and costly)
  - annotation is multifunctional and can be used for different purposes
  - records the linguistic analysis explicitly - possible to criticize
  - annotated corpus is used as standard reference resource

Corpus annotation

- Criticism:
  - corpus annotation produces cluttered corpora - it is important to see the plain text - question of representation
  - annotation imposes a linguistic analysis upon a corpus user - provides an objective record of an explicit analysis
  - accuracy and consistency of corpus annotation - automatic annotation is noisy - even the best linguists make mistakes

Annotation

- metadata, extra textual info (e.g. title, chapter, author, year)
- structural annotation (e.g. sentences, words)
- linguistic annotation
  - part-of-speech, morphological by tagging
  - lemma i.e., base form of each word
  - syntactic analysis by parsing - parsed corpora = treebank
  - text linguistic annotation
  - semantic information
- alignment on sentence and possibly word level (parallel corpora)

Principles for annotation (Leech 1993)

- An annotated corpus should be reversible to its original unannotated form.
- The annotation should be possible to be extracted from the text into a separate file.
- The annotation scheme should be documented and available for the user.
- It should be clear how the annotation was done and by whom.
- The user should be informed about that the annotation is not "God's truth" but a useful tool.
- The annotation scheme should be based upon accepted and theory-neutral principles.
- No annotation scheme is standard from the beginning. Standards are developed by the community.

Grammatical annotation

- Input: Running text
- Morphological segmentation, lemmatisation (start-ed, start)
- Part-of-speech tagging: to annotate tokens with their correct PoS (start/V)
- Chunking: to find non-overlapping group of words (NP: a nice journey PP: to NP: Uppsala)
- Syntactic parsing: to recover the complete syntactic structure

Terminology

- Key terms: token, type, hapax, lemma

  Token: sequences of letters separated by spaces or punctuation

  Type: counting each repeated item once

  Hapax legomena: the word that occur only once

  Lemma: base word form

Example

- How many tokens and types can you find in the text below?

  It is no exaggeration to say that copora, and the study of corpora, have revolutionized the study of language, and of the applications of language, over the last few decades.

Example: answer

It is no exaggeration to say that copora, and the study of corpora, have revolutionized the study of language, and of the applications of language, over the last few decades.

- Tokens: 35 including punctuation marks
- Types: 20

Preparing text for annotation

- Grammatical annotations are usually added to words and also to punctuation marks (period, comma)
- Tokenisation (1)
  - segmenting running text into words/tokens and
  - separating punctuation marks from words
  - white space marks token boundary, but not sufficient even for English: 'Book that flight!', he said.
  - Treat punctuation as word boundary: ' Book that flight ! ' , he said .

Tokenization

- Punctuation often occurs word internally
- Examples: Ph.D., google.com, abbreviations (e.g.), numeral expressions: dates (06/02/09), numbers (25.6, 100,110.10 or 100.110,10)
- Clitic contractions marked by apostroph: we're - we are
- Apostroph also as genitive case marker: book's
- Multiword expressions (White house, New York, etc) can be handled by a tokenizer by using a multiword expression dictionary - Named Entity Recognition (NER)

Preparing text for annotation

- Grammatical annotation is usually carried out on the sentence level
- Sentence/utterance segmentation (1)
  - segmenting a text into sentences is based on punctuation
  - certain kinds of punctuation (period, question mark, exclamation point) tend to mark sentence boundary
  - relatively unambiguous markers: ?, !

Preparing text for annotation

- Sentence/utterance segmentation (2)
  - Problematic: period as ambiguous between sentence boundary marker and a marker of abbreviations (Mr.) or both (This sentence ends with etc.).
  - Disambiguating end-of-sentence punctuation (period, question mark) from part-of-word punctuation (e.g., etc.)
  - Sentence segmentation and tokenization tend to be addressed jointly

Preparing text for annotation

They

neither

liked

nor

disliked

the

Old

Man

.

The

...

Part-of-Speech (PoS) tagging

- Goal: to assign each word a unique part-of-speech
- CONtent/N or conTENT/A (e.g. TTS, SR, parsing, WSD)
- PoS: noun, verb, pronoun, preposition, adverb, conjunction, participle, article, ...
- Tagset: a tag represents PoS with or without morphological information
  - 87 tags in Brown corpus (Francis, 1979)
  - 45 tags in Penn Treebank (Marcus et al., 1993)

Part-of-speech tagging

- Example:
- The/DT grand/JJ jury/NN commented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.

- Input: string of words and a specified tagset
- Output: single best tag for each word

Part-of-speech tagging, cont.

- Trivial
  - non-ambiguous words
- Non-trivial:
  - resolve ambiguous words (more than one possible PoS)
    - Book/VB that/DT flight/NN ./.
    - book NN VB
    - that DT CS
  - unknown words not present in the training data

Types of tagger

- Rule-based
  - Earliest taggers (Harris, 1962; Klein and Simmons, 1963; Green and Rubin, 1971)
  - Two-stage architecture:
    1. Use a dictionary to assign each word a list of potential PoS
    2. Use large lists of hand-written disambiguation rules to assign a single PoS for each word
  - The dictionaries and the set of rules get larger
  - Ambiguities often left unsolved in case of uncertainty

Data-driven tagging

- Goal: each word recieves a unique PoS (no ambiguities left)
- Usual steps in tagging:
  - Input: text/transcribed speech
  - Lexicon lookup: tagging with "default" tags
  - Disambiguation of ambiguous words
  - Output: Each word is annotated with one PoS tag

Data-driven taggers

- requires data set (supervised training)
- learning: algorithm to find the best explanation for the observation in a corpus
- classification problem (discret classes)

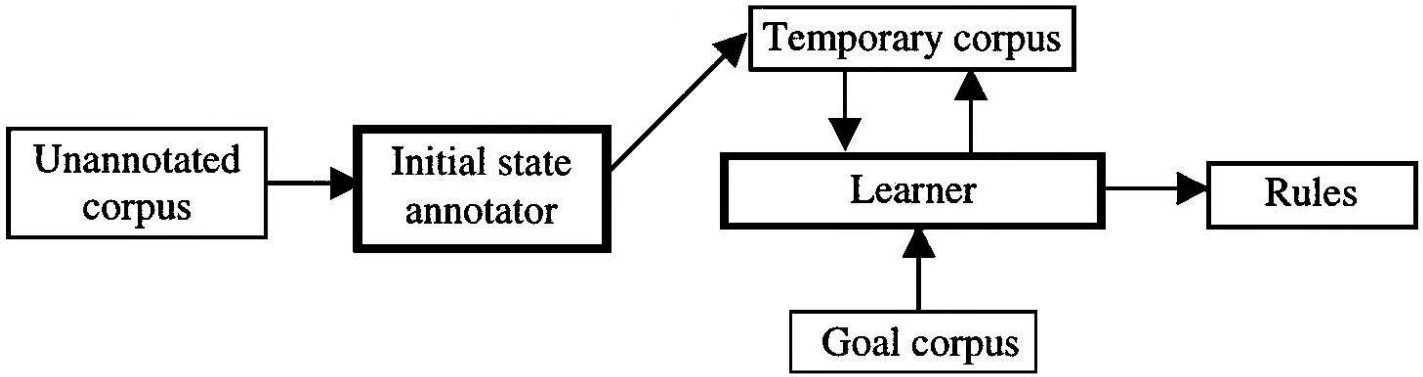
To decide

- algorithm/learning method to use
- represent the class (tagset)
- attributes to use (linguistic analysis)
- data size
  - training set
  - validation set
  - test set
- evaluation method

Transformation-based learning (TBL)

- Eric Brill 1992, 1995
- also called Brill tagger
- one of the first popular data-driven taggers
- based on rules (or transformations) which determine when ambiguous words should have a given tag
- ML component: grammar rules are automatically induced from a tagged training corpus
- system learns by detecting errors

TBL


Unannotated corpus
Initial state annotator
Temporary corpus
Learner
Rules
Goal corpus

Stochastic taggers

- Resolve ambiguities by using a training corpus to compute the probability of a given word having a given tag in a given context
- Hidden Markov Model or HMM tagger
  - HMM tagging is a task of choosing a tag-sequence with the maximum probability
  - Tagging is treated as a sequence classification task:
    - What is the best sequence of tags which corresponds to a particular sequence of words?

How an HMM tagger works?

- Compute tag frequencies for each tag
- Calculate the word likelihood probabilities, $P(word_i|tag_i)$, represent the probability, given that we see a given tag associated with a given word, i.e., we compute lexical frequencies by PoS-category for each word
- Calculate the tag sequence probabilities $P(t_i|t_{i-1})$ (bigram frequencies)
- Calculate products of lexical likelihood and tag sequence probabilities and decide the PoS tag.

The most probable tag sequence

- Secretariat/NNS is/VBZ expected/VBN to/TO race/VB tomorrrow/NR
- Example: race / VB or NN?
- NNS VBZ VBN TO VB NR
- NNS VBZ VBN TO NN NR
- Ambiguity resolves globally (not locally) picking the best tag sequence for the whole sentence

Computing the most probable tag sequence

- What is the likelyhood that the word race has VB and NN tag?

$$P(w_i|t_i)$$

- We can derive the probabilites (lexical likelihoods) from corpus counts:
- $P(race|NN) = \frac{C(race,NN)}{C(NN)} = .00057$ (How likely that the noun is *race*?)
- $P(race|VB) = \frac{C(race,VB)}{C(VB)} = .00012$ (How likely that the verb is *race*?)

Computing the most probable tag sequence

- How likely are we to expect a verb/noun given the previous tag?

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}$$

- We can derive the maximum likelihood estimate of a tag transition probability from corpus counts:
- $P(NN|TO) = \frac{C(TO,NN)}{C(TO)} = .00047$
- $P(VB|TO) = \frac{C(TO,VB)}{C(TO)} = .83$

Computing the most probable tag sequence

- What is the tag sequence probability for the following tag (tomorrow/NR)?

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}$$

- We can derive the probabilites from corpus counts.
- $P(NR|VB) = \frac{C(VB,NR)}{C(VB)} = .0027$
- $P(NR|NN) = \frac{C(NN,NR)}{C(NN)} = .0012$

Computing the most probable tag sequence

- Putting together the results:

$$\underset{t_1^n}{\operatorname{argmax}} \prod_{i=1}^{n} P(w_i|t_i)P(t_i|t_{i-1})$$

- $P(VB|TO)P(NR|VB)P(race|VB) =$ .83*.00012*.0027 = .00000027
- $P(NN|TO)P(NR|NN)P(race|NN) =$.00047*.0012*.00057 = .00000000032
- The prob of the sequence with the VB tag is higher and *race* is tagged as VB although it is less likely for *race*.

Tagset size

- depending of the corpus and language type
- tagset size for English: 50 - 100 tags
- tagset size for Swedish: SUC - 167 tags
- for agglutinative and highly inflectional languages, the tagset size is much larger as they are sequences of morphological tags rather than a single tag

Tagset size

- Comparisons in the morphologically tagged MULTEXT-East corpora (Hajic, 2000)

Language	Tagset size
English	139
Czech	970
Estonian	476
Hungarian	401
Romanian	486
Slovene	1033

Syntactic annotation

- Phrase structure grammar (Chomsky 1956-)
  - Constituents: NP, VP, AP, AdvP, PP
  - based on a hierarchical constituent structure
- Dependency grammar (Tesniere, Melcuk, 1950-)
  - DG is based on binary syntactic and semantic relations between words in a sentence (head and dependents)
  - Grammatical relations: subj, obj, dat, advl, attr, mod, ...
  - the finite verb is the head of the sentence
  - good for languages with relatively free word order (Czech)

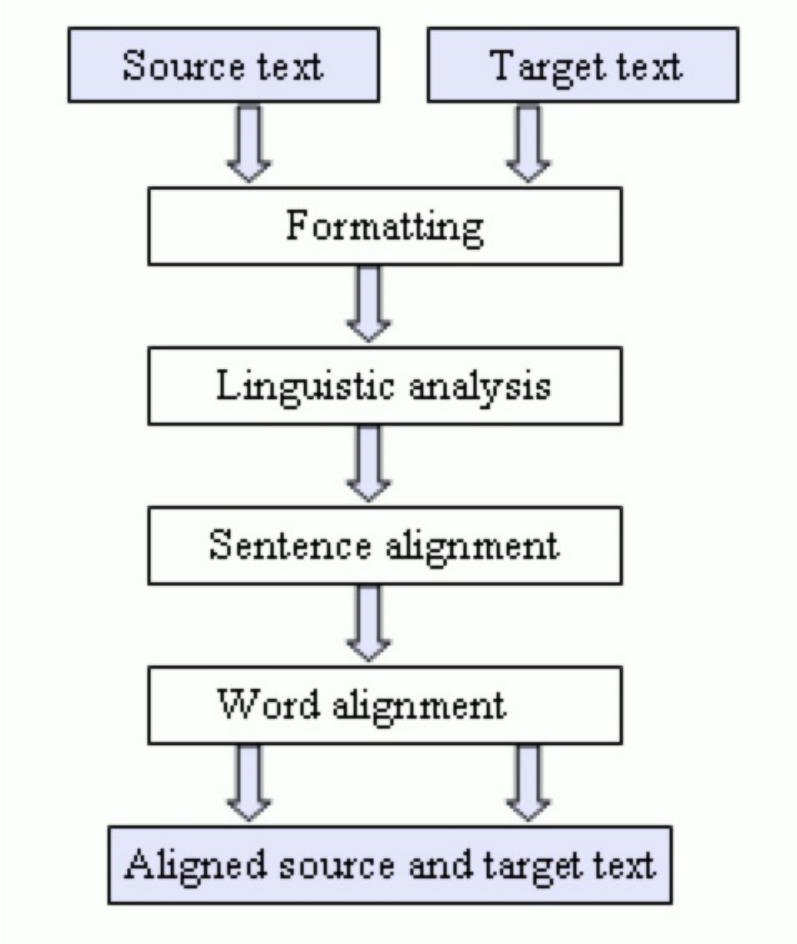
Syntactic annotation: Parsing

- Syntactic parsing:
  - construct the whole constituent or dependency structure for the sentence/utterance
- S[NP[[The AP[late] train PP[[from] NP[Stockholm]]] VP[has arrived]]

Syntactic annotation: Chunking

- a subproblem to syntactic parsing
- find syntactically related non-overlapping group of words, chunks (Abney, 1991): NP, VC, PP, ADVP, AP
- base phrase includes the headword of the phrase with pre-head words within the constituent
- no post-head words (removes the need to resolve attachment ambiguities)
- NP[The late train] PP[from] NP[Stockholm] VP[has arrived]
- NP[The late train] from NP[Stockholm] has arrived

Lab: Parallel corpus creation



Next time

- Treebanks and parallel corpora
- Language studies
- Project work: Think about a topic
  - Carry out an empiricial language study given a corpus of your choice: lexical study, grammatical structure, etc.
  - Build your own small corpus for a particular purpose