Corpus Linguistics
(L615)
Application #4: Translation

Markus Dickinson
Department of Linguistics, Indiana University
Spring 2013
Translation: Applications

Corpus-based translation studies

- **Theory**: How is an idea from one language conveyed in another language?
  - Compare different linguistic features in comparable texts
- **Practice**: provide material for:
  - Training translators
  - Developing applications like machine translation (MT) and computer-assisted translation (CAT)
Multilingual corpora are corpora with multiple languages (two or more)

- Gain new insights, as compared to monolingual corpora
- Highlight language-specific, typological, or cultural features
- Useful for lexicography
Multilingual corpora (cont.)

Three types of multilingual corpora:

▶ Type A: Source texts plus translations (e.g., Hansards)
▶ Type B: Monolingual subcorpora designed with the same sampling technique
▶ Type C: Combination of A&B (e.g., EMILLE)

**Parallel corpus** (for us) is Type A, and **comparable corpus** is Type B

▶ Corpora with different varieties of the same language (e.g., Brown, LOB) are **comparative corpora**
Parallel corpora

Parallel corpora can be uni- or multi-directional
  ▶ i.e., there could be translations in either or both directions

Parallel corpora are useful for a variety of purposes
  ▶ e.g., comparing the scopes of meaning of the progressive in Chinese & English
  ▶ e.g., training MT models for word translations
    ▶ MT training is much harder with, e.g., comparable corpora

The direction of translation is particularly important, as it may affect the naturalness
In general, parallel corpora are limited by the direction of the translation

- Each translation represents only one person’s interpretation
- This affects corpus searching and dramatically affects MT evaluation
  - i.e., there is no one “gold standard”

Also, a translation may contain distinctive features not found in regular use of that language, e.g.:

- Relatively lower proportion of lexical words over function words
- Relatively higher proportion of higher-frequency words
- Less variety in the words that are most frequently used

One solution is to use bidirectional corpora
Comparable corpora for translation

It turns out that corpus-aided translation has benefits

- Higher quality w.r.t. subject field understanding, correct term choice, & idiomatic expressions
- Translators seem to make fewer mistakes
  - especially when translating from a mother tongue into a foreign tongue
Machine translation (MT)

Machine translation relies upon parallel corpora

- Example-based machine translation (EBMT): compare a new sentence to a database of aligned texts
- Statistical machine translation (SMT): learn parameters from a parallel corpus
Alignment

An important step is to *align* corpus units

- at the level of the text, section, paragraph, sentence, and/or word
- often useful to have a separate alignment file with pointers to, e.g., word IDs

Basic methods for doing sentence alignment automatically:

- statistical: based on sentence length, in terms of words or characters
- lexical/rule-based: exploit morpho-syntactic information to align
  - often more accurate, but slower, than statistical approaches
- hybrid: integrate linguistic knowledge into a probabilistic system

Fairly accurate for sentence alignment of European language pairs
Recent trends in cross-linguistic lexical studies
Altenberg & Granger 2002

Contrastive linguistics characterized by:
▶ description of items to be compared
▶ juxtaposition of cross-linguistic equivalents
▶ comparison between the items

Altenberg & Granger are interested in defining equivalence across languages

Parallel corpora are useful for general definitions of word equivalence:
▶ There are different ways of defining equivalence between words
  ▶ translation equivalence seems the most reliable
▶ Judgment is not up to the researcher with translator equivalence
Inconsistencies in translation?

One way to deal with inconsistencies with different translations in determining equivalence:

▶ Only examine words which lead to the same back-translation
▶ This could eliminate translator idiosyncracies
  ▶ Quantitatively, we can measure *mutual correspondence* between words: how often is word X translated as word Y, and vice versa?
Using comparable corpora, we can compare aspect marking across languages

- Aspect marking in English & Chinese
  - perfective aspect in Chinese marked by -le, -guo, verb reduplication, & resultative verb complements
  - imperfective aspect in Chinese marked by zai, -zhe, -qilai, & -xiaqu
  - In English: imperfect indicated by progressive & perfect progressive

- Both languages show more aspectual marking in narrative texts
Some notable parallel corpora

- **MULTEXT-East**: for Bulgarian, Croatian, Czech, English, Estonian, Hungarian, Lithuanian, Resian, Romanian, Russian, Slovene, and Serbian. For most languages: Orwell’s 1984.

- **Hansard Corpus**: from the official records (Hansards) of the 36th Canadian Parliament [1997-2000], 3 mio. words

- **Europarl**: extracted from the proceedings of the European Parliament; includes versions in 11 European languages: Romanic (French, Italian, Spanish, Portuguese), Germanic (English, Dutch, German, Danish, Swedish), Greek and Finnish. Ca. 20 mio. words.

http://en.wikipedia.org/wiki/Parallel_text lists a few more
Working with multilingual corpora

On miller at /Volumes/Data/multilingual/

Let’s look specifically at the EUROPARL corpus (europarl_v7/, http://www.statmt.org/europarl/):

▶ parallel corpus extracted from the European Parliament web site by Philipp Koehn
  ▶ v.6: 21 European languages; roughly 60 million words per language
▶ main intended use is to aid statistical machine translation research
Sentence aligner

From the READMEv2:

Sentence aligner
----------------
You can create any parallel corpus with the command

```
./sentence-align-corpus.perl L1 L2
```

where L1 and L2 can be any of the 11 languages
```
da de el en es fi fr it nl pt sv
```

The output is stored in the aligned/ directory
...
Creating a parallel corpus takes about half an hour
on a 2GHz Linux machine.
We have the English & German subcorpora aligned
  ▶ found in the aligned/de-en/ subdirectory

The text is broken up with meta-information
  ▶ <CHAPTER> tags indicate a new chapter
  ▶ <SPEAKER> tags indicate when a new person is speaking
  ▶ <P> marks paragraph boundaries

We can use these coarse structural boundaries to help us track down equivalencies we are interested in
Translational for *one*

**Task:** interested in translations for *one* from English to German

1. Search the English corpora for all instances of *one*
   - Mark the structural unit in which each one occurs
2. Search the corresponding German corpora, looking for the specific locations where *one* occurred
   - Output those units in both English & German
   - If we knew all possible translational equivalents, we could restrict our attention to sentences of interest
Implementation
Searching the English corpus

We loop over all the files in the English directory: see the perl code (one.pl) for more details on that.

We have 3 main bookkeeping variables:

$unitnum = 1;  # which structural unit we’re on
$found_one = 0;  # whether we’ve found 'one' or not
$previous_lines = "";  # all the lines to print out
We search each line for the presence of *one* and keep track of every line within this structural unit:

```perl
# Store all the previous lines
$previous_lines .= $_;
}

# Test whether 'one' appears in this unit: if so, set $found_one to be some non-zero value
if (/one\b/i) {
    $found_one = $unitnum;
}
```
The corpus is aligned by structural units, such as paragraph and speaker.

Thus, we define a simple function which tells whether we’ve hit one of those markers:

```perl
sub new_unit {
    if (/<(CHAPTER|SPEAKER|P)>)/) { return 1; }  
    else { return 0; }
}
```

Then, if we hit a new unit and if we also have previously found one, we want to record that (see next slide).
Implementation
Indexing the items

We store the index of this item + its content in a dictionary En

# new_unit indicates that we’re starting
# a new unit of text
if (&new_unit($_)) {

# The criterion for adding to the En hash is that
# we’ve found the word we’re looking for
if ($found_one) {
    # make the original word more noticeable
    $previous_lines =~ s{\b(one)\b}{<ONE>$1</ONE>}ig;

    # store this position for later
    $key = $filename.'='.unitnum;
    $En{$key} = $previous_lines;
}

}
The search through the German corpus is similar, but the criterion is no longer whether we’ve seen *one*

- Rather: is this the same structural unit as in the English corpus?

```perl
if (&new_unit($_)) {
    $key = $filename.'='.'$unitnum;
    # The criterion for adding to the De hash is that
    # the corresponding English unit had the word
    # we’re looking for
    if (exists $En{$key}) {
        $De{$key} = $previous_lines;
    }
}
...
```
Implementation

Outputting the items

```perl
@en_keys = keys %En;
foreach $key (@en_keys) {
    if (exists $De{$key}) {
        select EN_OUT;
        print "$key\n$En{$key}\n";

        select DE_OUT;
        print "$key\n$De{$key}\n";
    }
}
```
The existence of a panoply of fundamental civil, economic, social and political rights, fully guaranteed by constitutional legislation and by public authorities, is one of the essential elements of modern Western society.
Das Vorhandensein einer durch die verfassungsmaßige Ordnung und die öffentlichen Gewalten umfassend garantierten Sammlung bürgerlicher, wirtschaftlicher, sozialer und politischer Grundrechte ist eines der wesentlichen Elemente der modernen westlichen Zivilisation.
Using the sentence alignment

We are currently just using the structural markers

- What do we need to do to make this find the specific aligned sentences within the files?
Taking it further
SMT

If you want to do more fine-grained alignment & the corpus doesn’t offer such annotation:

- Try using some modern statistical MT tools which do automatic alignment
- e.g., Joshua (http://joshua-decoder.org)
Taking it further
Visualization

For visualization & other tools, you can consult, e.g., this (out-of-date) website:

- http://www.cse.unt.edu/~rada/wa/

One general-purpose search & visualization tool is ANNIS: http://www.sfb632.uni-potsdam.de/annis/

- fyi: This requires installing PostgreSQL (http://www.postgresql.org/download/)
ANNIS2

Init database

Import corpus

List imported corpora

Annis started

Launch Annis frontend

Exit
ANNIS2
Querying alignments
Displaying other annotations