ISMLA Multilingual Session 5: Analysing the Subjunctive in French and Spanish

Johannes Dellert

Tübingen University

December 6, 2017

Plan

1. The Subjunctive Mood
2. How Similar are the Subjunctives of Romance Languages?
3. Defining the Concordance
4. Exercise 04: The Subjunctive in French and Spanish

The Subjunctive Mood

Today we will demonstrate the use of DKPro as a tool for exploring a linguistic question:

- **subjunctive**: an irrealis mood for expressing different kinds of unreality (opinions, obligations, wishes, ...)
- actual usage differs from language to language
- prototypical usage: in complement clause to verbs of demanding
  - `eng`: *I demand that he* ***come*** *tomorrow.*
  - `deu`: *Ich verlange, dass er morgen kommt.* (älter: ***komme***)
  - `fra`: *Je demande qu'il* ***vienne*** *demain.*
  - `spa`: *Exijo que* ***venga*** *mañana.*
  - `pol`: *Żądam, żeby* ***przyszedł*** *jutro.*
- German terminology: *Konjunktiv*
- mainly of interest: languages where the subjunctive is different from the conditional (in German: both *Konjunktiv*)

Subjunctives in Romance Languages

All of the four major Romance languages make this distinction:

- `fra`: *subjonctif* vs. *conditionnel*
- `spa`: *subjuntivo* vs. *condicional*
- `por`: *subjuntivo* vs. *condicional*
- `ita`: *congiuntivo* vs. *condizionale*

NB: The common ancestor Latin did **not** make the distinction!

Subjonctif: French examples

Je	croi-s	qu'	il	le	sai-t.
1SG	believe-1SG	that	3SG	3SG.ACC	know-3SG

"I believe that he knows it."

Je	ne	croi-s	pas	qu'	il	le	sache.
1SG	NEG	believe-1SG	step	that	3SG	3SG.ACC	know.SBJV.3SG

"I don't believe that he knows it."

Subjuntivo: Spanish examples

Ve-o	que	fuma-s.
see-1SG	that	smoke-2SG

"I see that you smoke."

Me	molesta	que	fume-s.
1SG.ACC	bother.3SG	that	smoke.SBJV-2SG

"It bothers me that you smoke."

Research Questions

Comparing the usage of the subjunctive mood in French and Spanish leads us to a series of exploratory questions:

1. Is the subjunctive used with equal frequency in both languages?
2. Is it possible (and common) for subjunctives to occur outside subordinate clauses? Do the languages differ in this?
3. Are there subjunctions which trigger the subjunctive more frequently in one language than their equivalent in the other?
4. Which role does negation play in the choice of indicative vs. subjunctive? Are there language-specific differences?

Data-Driven Comparison: Approach

Our approach to answering such questions in a data-driven way:

- extract all instances of the subjunctive from a parallel text
- summarize the instances and their context as a structured **concordance** (one instance per line, plus relevant context)
- classify the instances based on structural features of the context
- compare the numbers of instances in different contexts

In a real study, we would want more reliable tools for all of these steps!

Relevant Context for Subjunctives

All the relevant context information **precedes** the subjunctive verb form:

- is the form negated or not?
- which **subjunction** (subordinating conjunction) governs the subjunctive form, if any?
- which verb in the matrix clause governs the subjunction?
- is the relevant verb in the matrix clause negated or not?

Simplification

Because we do not have access to good dependency trees
(the output of DKPro tools with the pre-trained models is rubbish),
we **simplify these questions** to the level of pos tag sequences:

- is the form immediately preceded by an adverb meaning “not”?
- which subjunction is the first one we meet when moving to the left?
- in the matrix clause outside (i.e. to the left of) the subjunction, which verb form do we meet first?
- is the matrix verb immediately preceded by an adverb meaning “not”?

Formal Description of Concordancer

Summing up, we want to build concordances of the following pattern:
((NEG) V .* SBJ) .* (NEG) VS

- NEG: negation adverb (*no* in Spanish, *ne* or *n'* in French)
- SBJ: a subjunction (*que* “that” in both languages often erroneously analyzed as a pronoun)
- V: any verb form
- VS: a verb form in subjunctive mood
- .*: any number of intervening words (non-greedy)
- (): brackets express optionality

Exercise 04: The Data

- we will work with the French original and the Spanish translation of *Le roman d'un jeune homme pauvre*, a novel by Octave Feuillet (1858) ("The Story of a Poor Young Man")
- still counted as a classic a hundred years ago,
  but is mostly forgotten now (unlike some of Feuillet's plays)
- both texts (from Project Gutenberg), alongside an English translation, are packaged as `jeune-homme-pauvre.tar.gz`
- each file can be read in using
  `de.tudarmstadt.ukp.dkpro.core.io.text.TextReader`,
  i.e. it will not be necessary to implement a collection reader

Exercise 04: The DKPro Toolchain for French

Of all combinatory possibilities in DKPro, the following combination appears to work best:

`TextReader` with `TextReader.PARAM_LANGUAGE` set to `"fr"`

```
de.tudarmstadt.ukp.dkpro.core.stanfordnlp.StanfordSegmenter
de.tudarmstadt.ukp.dkpro.core.stanfordnlp.StanfordPosTagger
```

Do not use any other combination of tools!

Exercise 04: The DKPro Toolchain for Spanish

The single one of many possibilities in DKPro which works out of the box:

`TextReader` with `TextReader.PARAM_LANGUAGE` set to `"es"`

```
de.tudarmstadt.ukp.dkpro.core.languagetool.LanguageToolSegment
de.tudarmstadt.ukp.dkpro.core.opennlp.OpenNlpPosTagger
```

Do not even attempt to use any other combination of tools!

Exercise 04: General Requirements for the Concordancers

Your concordancer(s) should

- use the tagging result to find all instances of subjunctive verb forms
- for each subjunctive form, match the ten tokens immediately preceding it to the previously defined pattern
- print each instance on a separate line into a text file (configurable by a parameter)
- line format: the matched elements in the pattern (possibly empty), separated by tab characters, plus the entire sentence (see next slide)

Output example (line continues on next slide):

```
ne	crois	pas	qu'	il la	rende
```

Exercise 04: Extracting the Entire Sentence

Instructions for extracting the sentence that a given token belongs to:

- to maintain high performance, create an index `sentPerToken` from tokens into sentences before iterating through the tokens: `JCasUtil.indexCovering(jcas,Token.class,Sentence.class)`
- access the first element in the list stored by the index for the token: `sentPerToken.get(token).iterator().next()`
- to keep the sentence on one line, you need to do a `replaceAll("\n", " ")` before printing it to the output file

Output example (end of previous line):

```
-- Je ne crois pas qu'il la rende malheureuse.
```

("I do not believe that he makes her unhappy.")

Exercise 04: Implementing FraSbjvConcordancer

Specifics of the French concordancer:

- detecting subjunctive forms:
  `token.getPos().getPosValue()` is `"VS"`
- detecting negation:
  `token.getPos().getCoveredText()` is `"ne"` or `"n'"`
  (no special tag for negation, category is `ADV`)
- detecting subjunctions:
  `token.getPos().getPosValue()` is `"CS"` or `"PROREL"`
  (`PROREL` means “relative pronoun”, necessary due to mistokenization)
- detecting any verb form:
  use `instanceof` with `Token` subtype `V` from DKPro API

Exercise 04: Implementing SpaSbjvConcordancer

Specifics of the Spanish concordancer:

- detecting subjunctive forms:
  `token.getPos().getPosValue()` is `"VAS"`, `"VMS"`, or `"VSS"`
- detecting negation:
  `token.getPos().getPosValue()` is `"RN"`
- detecting subjunctions:
  `token.getPos().getPosValue()` is `"CS"` or `"PR"`
  (`PR` means “relative pronoun”, necessary due to mistokenization)
- detecting any verb form:
  use `instanceof` with `Token` subtype `V` from DKPro API

Exercise 04: Running the Pipelines

- build separate pipelines for each language version
- for each language, the pipeline should minimally include
  - the `TextReader` with language and source parameters set
  - the correct segmenter (without setting any parameters)
  - the correct tagger (without setting any parameters)
  - your language-specific (or parametrized) Concordancer
- for running, use `SimplePipeline.runPipeline` this time

Exercise 04: Interpreting the Results

Inspecting the results:

- the TSV output format can conveniently be opened and inspected with OpenOffice Calc (or Excel)
- use the possibility to sort the entries by the values in each column!

Guiding questions for the interpretation:

- Overall frequency of the subjunctive in both languages?
- Which subjunctions are most frequent?
- Is the subjunctive common in main clauses (no subjunction)?
- Does negation play an obvious role?
- If you know French or Spanish: How did the simple pattern perform? Are there examples where dependency parsing would have helped?

Exercise 04: Implementation Hints

- there are many ways in which the pattern can be matched; one variant operates on a list filled by a call to `JCasUtil.selectPreceding`
- the two languages are syntactically very similar, you can reuse large amounts of code between both language versions
  (or even better: parametrize everything)
- for debugging, we recommend adding an additional
  `de.tudarmstadt.ukp.dkpro.core.io.conll.Conll2006Writer`
  as a second CAS consumer to the end of your pipelines

Exercise 04: Questions

Questions?