To tree or not to tree

Zdeněk Žabokrtský

November 6, 2020

EUROPEAN UNION
European Structural and Investment Fund
Operational Programme Research,
Development and Education

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

Troubles with choosing an annotation scheme: a case study on problematic corpus/treebank design decisions

Corpus annotation criticism

- some critics: an annotated corpus is worse than a raw corpus because of forced interpretations
  - one has to struggle with different linguistic traditions of different national schools
  - example: part of speech categories
- relying on annotation might be misleading if the quality is low (errors or inconsistencies)

Variability of PoS tag sets

Penn Treebank POS tagset (for English)

CC coordinating conjunction (*and*)
CD cardinal number (*1, third*)
DT determiner (*the*)
EX existential there (*there is*)
FW foreign word (*d'hoevre*)
IN preposition/subordinating conjunction (*in, of, like*)
JJ adjective (*green*)
JJR adjective, comparative (*greener*)
JJS adjective, superlative (*greenest*)
LS list marker (*1)*)
MD modal (*could, will*)
NN noun, singular or mass (*table*)
NNS noun plural (*tables*)
NNP proper noun, singular (*John*)
NNPS proper noun, plural (*Vikings*)
PDT predeterminer (*¡iż both¡/iż the boys*)
POS possessive ending (*friend's*)
PRP personal pronoun (*I, he, it*)
PRP$ possessive pronoun (*my, his*)
RB adverb (*however, usually, naturally, here, good*)
RBR adverb, comparative (*better*)
RBS adverb, superlative (*best*)
RP particle (*give up*)
TO to (*to go, to him*)
UH interjection (*uhhuhhuhh*)
VB verb, base form (*take*)
VBD verb, past tense (*took*)
VBG verb, gerund/present participle (*taking*)
VBN verb, past participle (*taken*)
VBP verb, sing. present, non-3d (*take*)
VBZ verb, 3rd person sing. present (*takes*)
WDT wh-determiner (*which*)
WP wh-pronoun (*who, what*)
WP$ possessive wh-pronoun (*whose*)
WRB wh-abverb (*where, when*)

Variability of PoS tag sets, cont.

Negra Corpus POS tagset (for German)

ADJA Attributives Adjektiv
ADJD Adverbiales oder prädikatives Adjektiv
ADV Adverb
APPR Präposition; Zirkumposition links
APPRART Präposition mit Artikel
APPO Postposition
APZR Zirkumposition rechts
ART Bestimmter oder unbestimmter Artikel
CARD Kardinalzahl
FM Fremdsprachliches Material
ITJ Interjektion
KOUI Unterordnende Konjunktion mit zu und Infinitiv
KOUS Unterordnende Konjunktion mit Satz
KON Nebenordnende Konjunktion
KOKOM Vergleichspartikel, ohne Satz
NN Normales Nomen
NE Eigennamen
PDS Substituierendes Demonstrativpronomen
PDAT Attribuierendes Demonstrativpronomen
PIS Substituierendes Indefinitpronomen
PIAT Attribuierendes Indefinitpronomen
PIDAT Attribuierendes Indefinitpronomen mit Determiner
PPER Irreflexives Personalpronomen
PPOSS Substituierendes Possessivpronomen
PPOSAT Attribuierendes Possessivpronomen
PRELS Substituierendes Relativpronomen
PRELAT Attribuierendes Relativpronomen
PRF Reflexives Personalpronomen
PWS Substituierendes Interrogativpronomen
PWAT Attribuierendes Interrogativpronomen
PWAV Adverbiales Interrogativ- oder Relativpronomen
PROAV Pronominaladverb
PTKZU zu vor Infinitiv
PTKNEG Negationspartikel
PTKVZ Abgetrennter Verbzusatz
PTKANT Antwortpartikel
PTKA Partikel bei Adjektiv oder Adverb
TRUNC Kompositions-Erstglied
VVFIN Finites Verb, voll
VVIMP Imperativ, voll
VVINF Infinitiv, voll
VVIZU Infinitiv mit zu, voll
VVPP Partizip Perfekt, voll
VAFIN Finites Verb, aux
VAIMP Imperativ, aux
VAINF Infinitiv, aux
VAPP Partizip Perfekt, aux
VMFIN Finites Verb, modal
VMINF Infinitiv, modal
VMPP Partizip Perfekt, modal
XY Nichtwort, Sonderzeichen
$, Komma
$. Satzbeendende Interpunktion
$(Sonstige Satzzeichen; satzintern
NNE Verbindung aus Eigennamen und normalen Nomen

Variability of PoS tag sets, cont.

Prague Dependency Treebank morphologitagset (for Czech), several thousand combinations using 15-character long positional tags

Form	Lemma	Morphological tag
Některé	*některý*	PZFP1----------
kontury	*kontura*	NNFP1-----A----
problému	*problém*	NNIS2-----A----
se	*se_^*(zvr._zájmeno/částice)	P7-X4----------
však	*však*	J^----------
po	*po-1*	RR--6----------
oživení	*oživení_^*(*3it)	NNNS6-----A----
Havlovým	*Havlův_;S_^*(*3el)	AUIS7M---------
projevem	*projev*	NNIS7-----A----
zdají	*zdát*	VB-P---3P-AA---
být	*být*	Vf--------A----
jasnější	*jasný*	AAFP1----2A----
.	.	Z:----------

Treebanks

- a treebank is a corpus in which sentences' syntax and/or semantics is analyzed using tree-shaped data structures
- a tree in the sense of graph theory (a connected acyclic graph)
- sentence syntactic analysis ... it sounds familiar to most of you, doesn't it?

Credit: http://konecekh.blog.cz

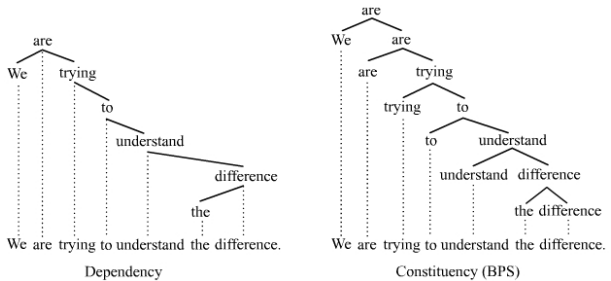
Why trees: Initial thoughts

1. Honestly: trees are irresistibly attractive data structures.
2. We believe sentences can be reasonably represented by discrete units and relations among them.
3. Some relations among sentence components (such as some word groupings) make more sense than others.
4. In other words, we believe there is an latent but identifiable discrete structure hidden in each sentence.
5. The structure must allow for various kinds of nestedness (*...a já mu řek, že nejsem Řek, abych mu řek, kolik je v Řecku řeckých řek ...*).
6. This resembles recursivity. Recursivity reminds us of trees.
7. Let's try to find such trees that make sense linguistically and can be supported by empirical evidence.
8. Let's hope they'll be useful in developing NLP applications such as Machine Translation.

So what kind of trees?

There are two types of trees broadly used:

- constituency (phrase-structure) trees
- dependency trees



Credit: Wikipedia

Constituency trees simply don't fit to languages with freer word order, such as Czech. Let's use dependency trees.

BTW how do we know there is a dependency between two words?

- There are various clues manifested, such as
  - word order (juxtapositon): "*...přijdu zítra ...*"
  - agreement: "*...novými*$_{.pl.instr}$ *knihami*$_{.pl.instr}$*...*"
  - government: "*...slíbil Petrovi*$_{.dative}$*...*"

- Different languages use different mixtures of morphological strategies to express relations among sentence units.

Basic assumptions about building units

If a sentence is to be represented by a dependency tree, then we need to be able to:

- identify **sentence boundaries**.
- identify **word boundaries** within a sentence.

Basic assumptions about dependencies

If a sentence is to be represented by a dependency tree, then:

- there must be a **unique parent word** for each word in each sentence, except for the root word
- there are **no loops** allowed.

Even the most basic assumptions are violated

- Sometimes **sentence boundaries are unclear** – generally in speech, but e.g. in written Arabic too, and in some situations even in written Czech (e.g. direct speech)
- Sometimes **word boundaries are unclear**, (Chinese, "ins" in German, "abych" in Czech).
- Sometimes its **unclear which words should become parents** (A preposition or a noun? An auxiliary verb or a meaningful verb? ...).
- Sometimes there are too many relations ("Zahlédla ho bosého."), which implies **loops**.

Life's hard. Let's ignore it and insist on trees.

Counter-examples revisited

If we cannot find linguistically justified decisions, then make them at least consistent.

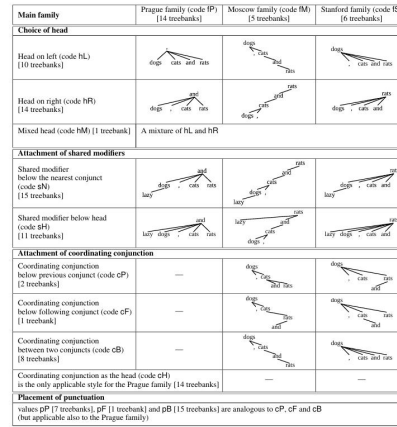
- Sometimes sentence boundaries are unclear (generally in speech, but e.g. in written Arabic too…)
  - **OK, so let's introduce annotation rules for sentence segmentation.**
- Sometimes word boundaries are unclear, (Chinese, "ins" in German, "abych" in Czech).
  - **OK, so let's introduce annotation rules for tokenization.**
- Sometimes it's not clear which word should become parent (e.g. a preposition or a noun?).
  - **OK, so let's introduce annotation rules for choosing parent.**
- Sometimes there are too many relations ("Zahlédla ho bosého."), which implies loops.
  - **OK, so let's introduce annotation rules for choosing tree-shaped skeleton.**

Treebanking

- Is our dependency approach viable? Can we check it?
- Let's start by building the trees manually.
- a treebank - a collection of sentences and associated (typically manually annotated) dependency trees
- for English: Penn Treebank [Marcus et al., 1993]
- for Czech: Prague Dependency Treebank [Hajič et al., 2001]
  - layered annotation scheme: morhology, surface syntax, deep syntax
  - dependency trees for about 100,000 sentences
- high degree of design freedom and local linguistic tradition bias
- different treebanks $\Longrightarrow$ different annotation styles

An example of a treebank variability cause: the case of coordination

- coordination structures such as "*lazy dogs, cats and rats*" consists of
  - conjuncts
  - conjunctions
  - shared modifiers
  - punctuation tokens
- 16 different annotation styles identified in 26 treebanks (and many more possible)
- different expressivity, limited convertibility, limited comparability of experiments...
- **harmonization of annotation styles badly needed!**



Btw how many treebanks are there out there?

- growing interest in dependency treebanks in the last decade or two
- existing treebanks for about 100 languages now (but roughly 7,000 languages in the world)
- UFAL participated in several treebank unification efforts:
  - 13 languages in CoNLL in 2006
  - 29 languages in HamleDT in 2011
  - 37 languages in Universal Dependencies in 2015:
  - 70+ languages in UD in 2019

Conclusion

- one should keep in mind that there's no straightforward "God's truth" when it comes to language data resources
- all resources are heavily influenced by numerous design choices, for which no perfect answers exists
- examples of trade-offs:
  - the bigger data the better, but you can't remove all noise from really big data
  - parallel annotation reduces the amount of annotation errors, but increases costs
  - linguistically-based annotation brings interpretability, but at the same time we risk being trapped in some suboptimal traditions that are possibly not useful beyond a given language family
  - a better quality/coverage is sometimes achievable by integrating more resources focused on a same task, but their licenses might be incompatible