cost
EUROPEAN COOPERATION
IN SCIENCE & TECHNOLOGY

UniDive Training School

Funded by
the European Union

Session 6

Basics of treebank querying and annotation

Bruno Guillaume, Sylvain Kahane

This is joint work with **Guillaume Bonfante**, **Guy Perrier**, **Kim Gerdes**, **Gaël Guibon**, **Kirian Guiller**, **Khensa Daoudi** and others

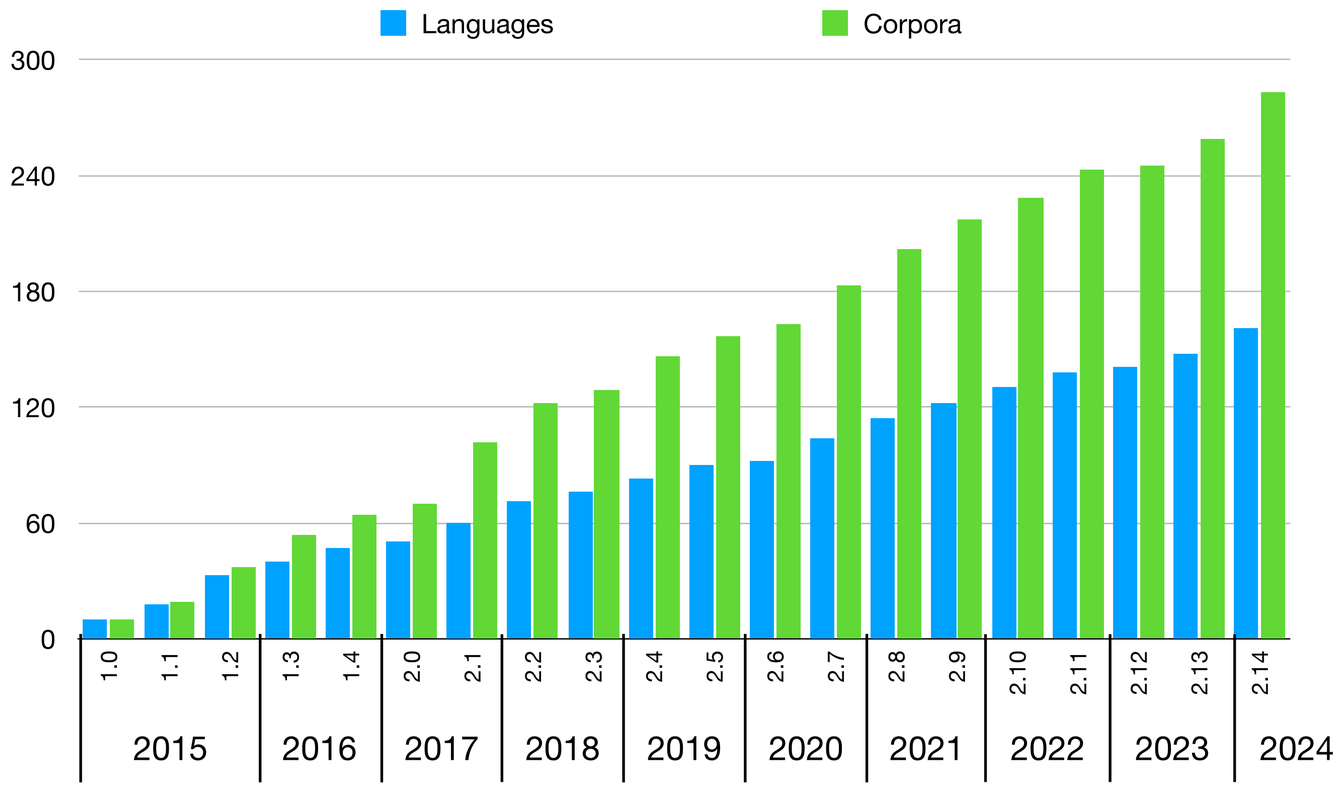
Outline

- Treebank exploration with **Grew-match**

- Treebank annotation and maintenance with **ArboratorGrew**

Universal Dependencies (UD)

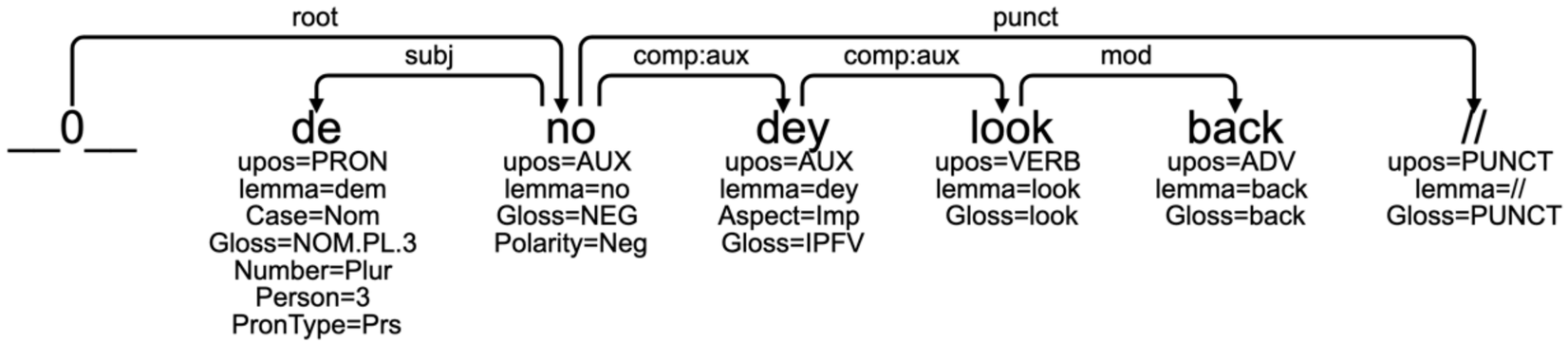
- Collaborative project of “universal” dependency annotations
- Version 2.14: 161 languages, 283 corpora

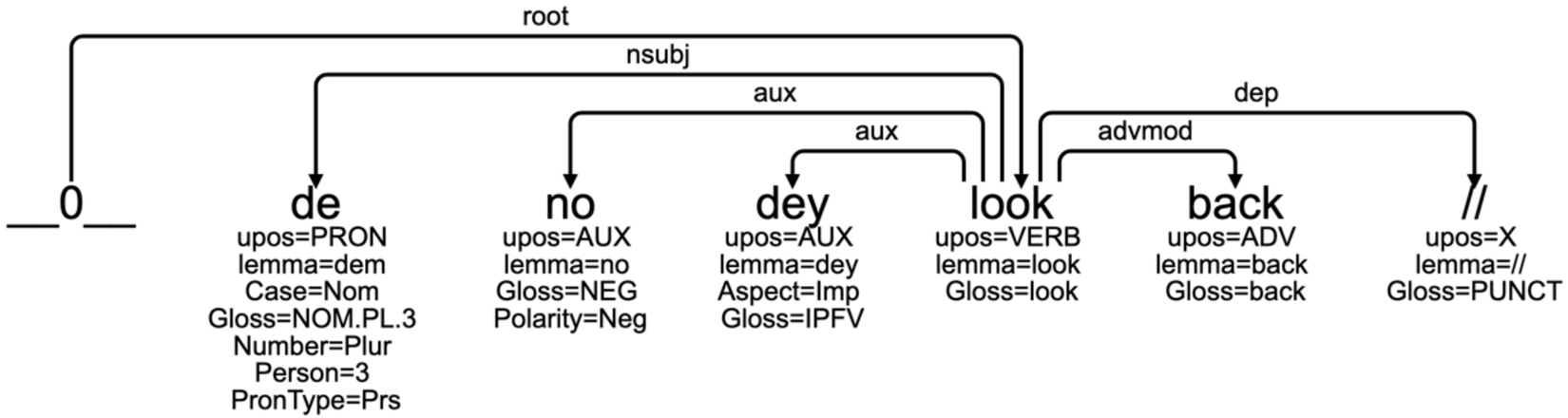


Surface Syntactic Universal Dependencies (SUD)

- Alternative way of designing dependency relations in sentence
- Parallel to UD for other annotation layers (tokenisation, POS, lemmas, morphology)

De no dey look back. (*They don't hesitate.*)





Treebank querying

Main portal: https://match.grew.fr

SUD_Naija-NSC: https://universal.grew.fr/?corpus=SUD_Naija-NSC@2.14

Explore POS annotation in SUD_Naija-NSC

Universal POS tags

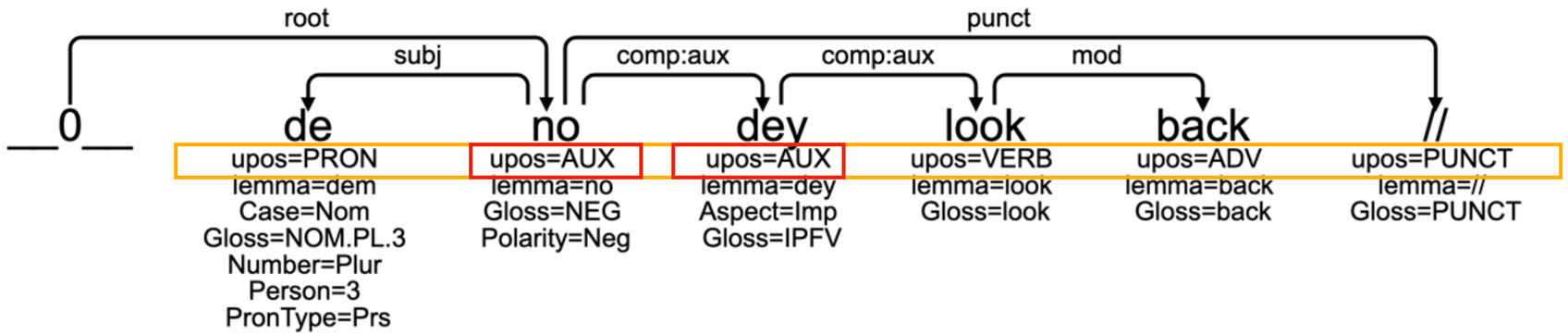
These tags mark the core part-of-speech categories. To distinguish additional lexical and grammatical properties of words, use the universal features.

Open class words	Closed class words	Other
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

UD guidelines:

https://universaldependencies.org/u/pos/index.html

De no dey look back. (*They don't hesitate.*)



Explore the AUX tag in SUD_Naija-NSC@2.14

Requests on Grew-match

Explore the AUX tag in SUD_Naija-NSC@2.14

```
pattern { X [upos = AUX] }
```

Search — More than 1000 results found in 7.67% of the corpus [0.388s]

Count — 10756 occurrences [0.099s]

Look for the lemmas of first 5 occurrences: *dey* (2), *no* (2) and *make* (1)

```
pattern { X [upos = AUX, lemma = "dey"] }
```

3128 occurrences [0.148s]

```
pattern { X [upos = AUX, lemma = "no"] }
```

1625 occurrences [0.091s]

```
pattern { X [upos = AUX, lemma = "make"] }
```

680 occurrences [0.093s]

What are the lemmas of the other ones?

```
pattern { X [upos = AUX, lemma<>"dey"|"no"|"make"] }
```

5323 occurrences [0.135s]

Found: *fin*, *con*…

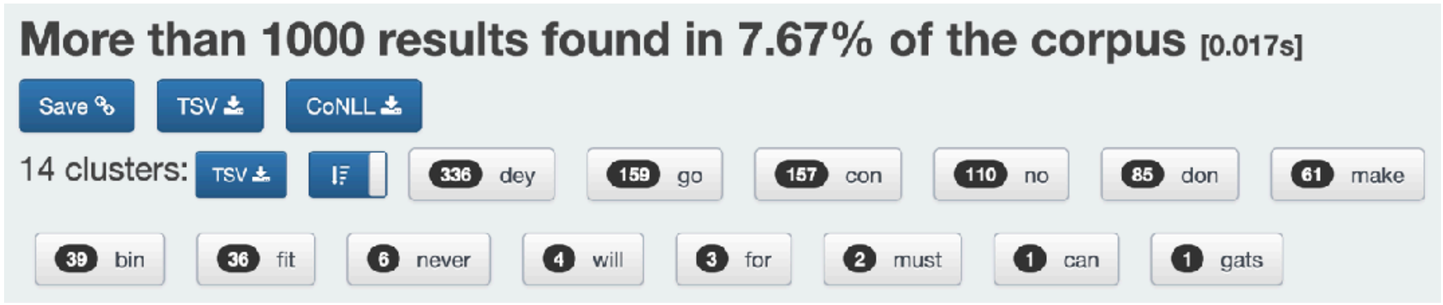
Requests with clustering

Questions like this one are very frequent:
for a given set of observations,
what is the distribution of some related observations?

X is an AUX
the lemma of X

In Grew-match, the **clustering** functionality does exactly this:

```
pattern { X [upos = AUX] }   X.lemma
```



Practise clustering

- Select a UD or SUD treebank for a language you know
  - List auxiliaries lemmas
  - For an ambiguous word, list possible POS for it
- For a language with a rich morphology https://match.grew.fr
  - Find the VERB lemma with the most occurrences
  - How many different forms can this lemma have?
    - Can you find more than ~~264 clusters~~?

FIX: add the VERB constraint ➔ 168 clusters

Requests on dependencies

Explore the **comp:obj** dependency relation in SUD_Naija-NSC@2.14

```
pattern { e: X -[comp:obj]-> Y }
```

18693 occurrences [0.126s]

```
pattern { e: X -[comp:obj]-> Y }   X.upos
```


```
pattern { e: X -[comp:obj]-> Y }   Y.upos
```

Practise

- For the most common AUX lemmas in one UD treebank
  - List the other POS which it can be used with
  - List the syntactic functions it has in the treebank
  - See the relation between POS and syntactic function
- On a SUD English treebank, observe the POS of the lemma *have* and the dependency relations it governs.
- On a UD English treebank, observe the POS of the lemma *have* and the dependency relations with its governor.

More with Grew-match

- Strict word order constraint: X comes immediately before Y `X < Y`
  - Search for n-grams (e.g. *in spite of* in English) 
- Non strict word order constraint: X comes before Y `X << Y`
  - Search for unexpected word order (e.g. VERB and its subj) 
- Clustering with a sub-pattern (whether)
  - Measure how an order is distributed (e.g. ADJ modifying a NOUN) 

    Yes ➔ ADJ is after the NOUN
    NO ➔ ADJ is before the NOUN
- *Negative* patterns `without { … }`
  - Search for verb without subject 

Practise

- For an object linked to a verb, observe the correlation between the position of the object and its POS in French and other Romance languages.
- Observe the relative position of the subject to its head, "explain" irregularities
- How can you check if a language is pro-drop?

Treebank annotation and maintenance

https://arborator.github.io/

Basic use of ArboratorGrew

- Connection
- Projects, Samples, Sentences, Users
- Basic edition of a tree
  - POS, lemma, morphology, MISC, dependency
- Metadata edition
- CoNLL view / edition
- Export data

Practise

https://arborator.github.io/

- Connect with Gmail or GitHub (prefer GitHub)
- Go to the project UTS_Naija
  - Annotate (part of) a sentence
  - Save your annotation
- Start a new projet (please name it UTS_…!)
  - Upload some data from https://github.com/UniDive/2024-UniDive-Chisinau-training-school/tree/main/Course-1-dependency-syntax/data

Advanced editing features in ArboratorGrew

Features are available only for validators and admins

- Tokenisation
- Sentence segmentation

Features are available only for admins

- Project settings
  - roles
  - features and metadata displayed in the annotation interface
  - configuration of available features and relations (JSON file)

Other features in ArboratorGrew

More on this later in the week

- Lexicons
- Parser access
- GitHub synchro