

Adding New Language to Stanza (Stanford NLP)

Reviewing the work required to achieve the goal

What is Universal Dependencies?

▶	 English	9	733K		IE, Germanic
▶	 Erzya	1	17K		Uralic, Mordvin
▼	 Estonian	2	511K		Uralic, Finnic

Estonian treebanks

▶	EDT	438K	LFD			★★★★☆
▶	EWT	72K	LFD			★★★★☆

See [here](#) for comparative statistics of Estonian treebanks.

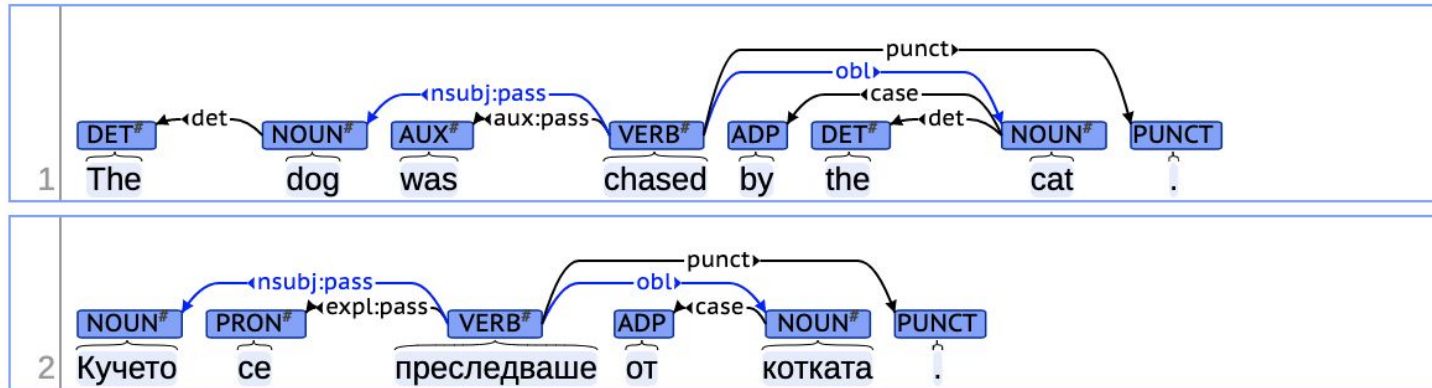
Language documentation

See the [language documentation page](#).










- Common place for language specific Treebanks
- CONLLU format annotations
- Well used for the standard nlp tasks: **tokenization, lemmatization, dependency parsing, named entity recognition**
- Well used for downstream language understanding tasks: **relation extraction, reading comprehension, machine translation** (using parallel treebanks)

How to Create Treebank for UD?

- Define full language grammar (<http://nl.ijs.si/ME/V6/msd/html/msd-ka.html>)
- Convert into the UD specification (<https://aclanthology.org/W15-1821.pdf>, <https://aclanthology.org/L16-1247.pdf>)
- Use annotation tools to label the sentences (<https://universaldependencies.org/tools.html#brat-rapid-annotation-tool>)



Example of UD Repository

 .gitignore	removed test from gitignore	5 years ago
 CONTRIBUTING.md	Updated CONTRIBUTING.md.	4 years ago
 LICENSE.txt	LICENSE.txt	6 years ago
 README.md	Update README.md	9 months ago
 et_edt-ud-dev.conllu	Add files via upload	9 months ago
 et_edt-ud-test.conllu	Add files via upload	9 months ago
 et_edt-ud-train.conllu	Add files via upload	9 months ago
 eval.log	Updated treebank evaluation.	3 months ago
 stats.xml	Updated statistics.	9 months ago

- Not important
- How to contribute (guide)
- License
- About
- Dev set
- Test set
- Train set
- Evaluation logs
- Stats

Example of CONLLU annotations

```
# sent_id = aja_ee199920_1969
# text = Palju olulisi komponente, nagu liha ja kala, hangime siiski Eestist.
1   Palju   palju   ADV     D           _           3           advmod  3:advmod  _
2   olulisi oluline ADJ     A           Case=Par|Degree=Pos|Number=Plur 3   amod     3:amod  _
3   komponente komponent NOUN    S           Case=Par|Number=Plur 10  obj      10:obj  SpaceAfter=No
4   ,        ,        PUNCT   Z           _           6           punct   6:punct  _
5   nagu     nagu     SCONJ   J           _           6           mark    6:mark   _
6   liha     liha     NOUN    S           Case=Gen|Number=Sing 3    appos    3:appos  _
7   ja       ja       CCONJ   J           _           8           cc      8:cc     _
8   kala     kala     NOUN    S           Case=Gen|Number=Sing 6    conj     6:conj   SpaceAfter=No
9   ,        ,        PUNCT   Z           _           10          punct   10:punct
10  hangime  hankima VERB    V           Mood=Ind|Number=Plur|Person=1|Tense=Pres|VerbForm=Fin|Voice=Act 0   root     0:root  _
11  siiski   siiski   ADV     D           _           10          advmod  10:advmod
12  Eestist  Eesti    PROPN   S           Case=Ela|Number=Sing 10   obl      10:obl   SpaceAfter=No
13  .        .        PUNCT   Z           _           10          punct   10:punct  _
```

- <https://universaldependencies.org/format.html>
- <https://universaldependencies.org/ext-format.html>
- <https://unimorph.github.io/>

What is possible to have in UD Treebank annotations?

Sentences consist of one or more word lines, and word lines contain the following fields:

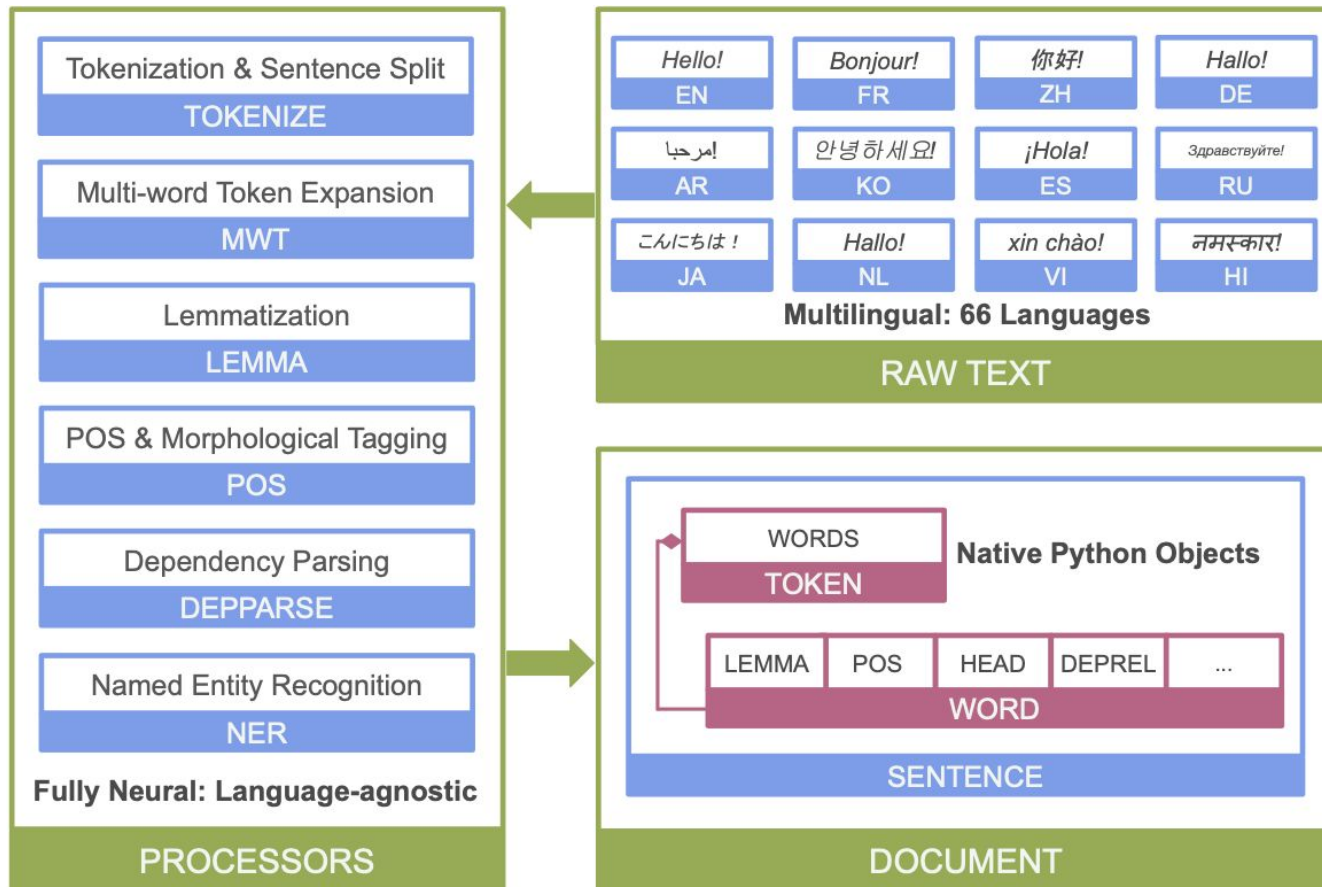
1. ID: Word index, integer starting at 1 for each new sentence; may be a range for multiword tokens; may be a decimal number for empty nodes (decimal numbers can be lower than 1 but must be greater than 0).
2. FORM: Word form or punctuation symbol.
3. LEMMA: Lemma or stem of word form.
4. UPOS: [Universal part-of-speech tag](#).
5. XPOS: Language-specific part-of-speech tag; underscore if not available.
6. FEATS: List of morphological features from the [universal feature inventory](#) or from a defined [language-specific extension](#); underscore if not available.
7. HEAD: Head of the current word, which is either a value of ID or zero (0).
8. DEPREL: [Universal dependency relation](#) to the HEAD ([root](#) iff HEAD = 0) or a defined language-specific subtype of one.
9. DEPS: Enhanced dependency graph in the form of a list of head-deprel pairs.
10. MISC: Any other annotation.

The fields DEPS and MISC replace the obsolete fields PHEAD and PDEPREL of the CoNLL-X format. In addition, we have modified the usage of the ID, FORM, LEMMA, XPOS, FEATS and HEAD fields as explained below.

The fields must additionally meet the following constraints:

- Fields must not be empty.
- Fields other than FORM, LEMMA, and MISC must not contain space characters.
- Underscore (`_`) is used to denote unspecified values in all fields except ID. Note that no format-level distinction is made for the rare cases where the FORM or LEMMA is the literal underscore – processing in such cases is application-dependent. Further, in UD treebanks the UPOS, HEAD, and DEPREL columns are not allowed to be left unspecified except in multiword tokens, where all must be unspecified, and empty nodes, where UPOS is optional and HEAD and DEPREL must be unspecified. The enhanced DEPS annotation is optional in UD treebanks, but if it is provided, it must be provided for all sentences in the treebank.

Stanza – A Python NLP Package for Many Human Languages



What's behind the Pipeline models of stanza

Tokenization and Sentence Split: On feeding raw text, Stanza tokenizes it and groups tokens into sentences as the first step of processing. Unlike other existing toolkits, Stanza combines tokenization and sentence segmentation from raw text into a single module. This is done to predict the position of words in a sentence, as use of words are context-sensitive in some languages.

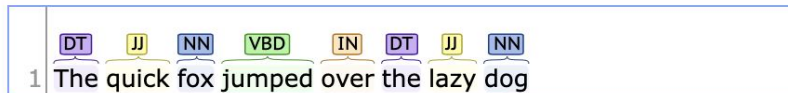
Multi-Word Token Expansion: The above methods identify multi-word tokens, which are then further extended into the syntactic words as the foundation for downstream processing. This is accomplished by the use of sequence-to-sequence (seq2seq) model to ensure frequently observed expansions in the training set, as they are always robustly expanded while maintaining the flexibility to model unseen words statistically.

Lemmatization: Stanza also lemmatizes each word in a sentence to regain its canonical form (e.g., did→do). Similar to the multi-word token expander, Stanza's lemmatizer is deployed as an ensemble of a dictionary-based lemmatizer and a neural seq2seq lemmatizer. Besides, an additional classifier is built on the encoder output of the seq2seq model, to predict shortcuts like lowercasing and identity copy for robustness on long input sequences such as URLs.

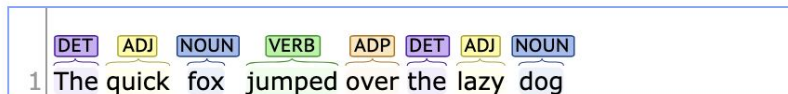
POS and Morphological Feature Tagging: For each word in a sentence, Stanza assigns it as a part-of-speech (POS), and evaluates its universal morphological features (UFeats, e.g., singular/plural, 1st/2nd/3rd person, among others). To predict POS and UFeats, researchers adopted a bidirectional long short-term memory network (Bi-LSTM) as the basic architecture.

Stanza Demo

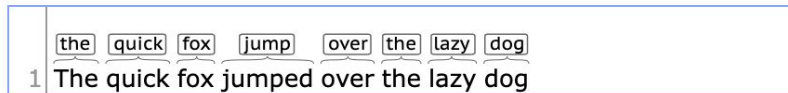
Part-of-Speech (XPOS):



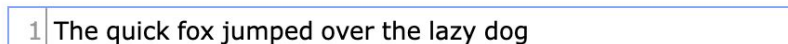
Universal Part-of-Speech:



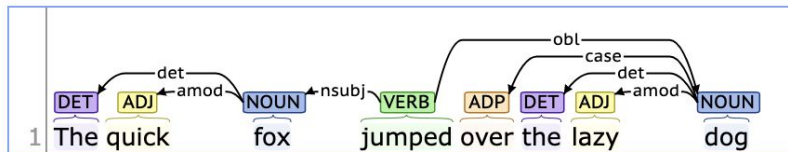
Lemmas:



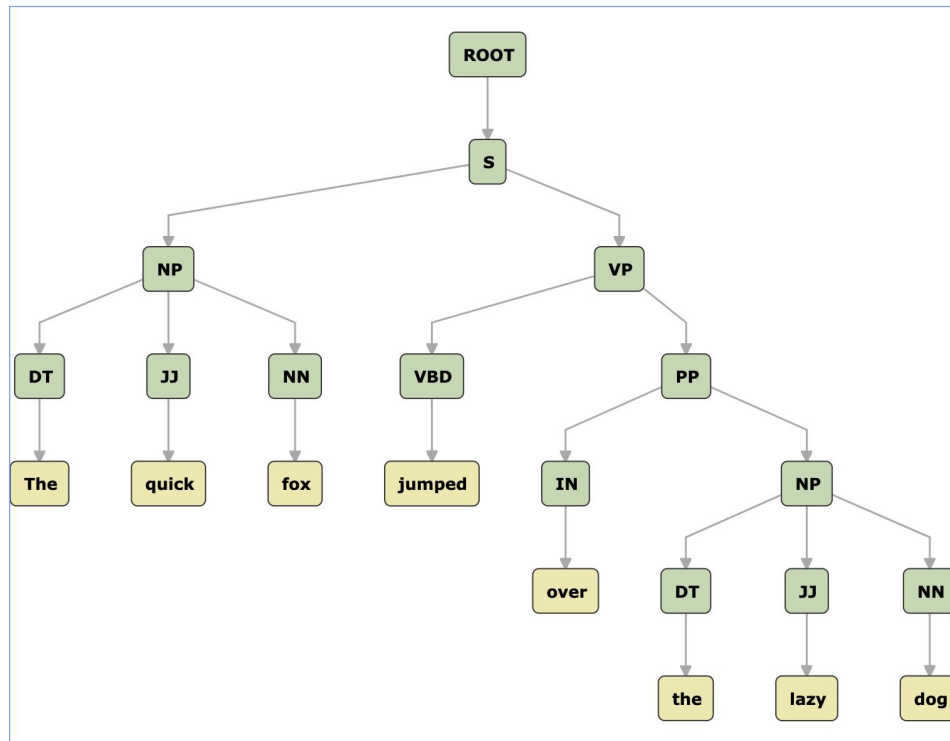
Named Entity Recognition:



Universal Dependencies:



Constituency Parse:



How to add new Language in Stanza?

- Generate Treebank (with Train, Dev, Test sets)
- Generate Word Vectors (word2vec, GloVe, etc)
- Train Stanza Models using UD dataset ([guide](#))
- Contribute (example of [issue](#) on Ukrainian NER)

Training Stanza Models on Sample Dataset

Barack	Barack	PROPN	NNP	Number=Sing	4	nsubj:pass	-	-		
Obama	Obama	PROPN	NNP	Number=Sing	1	flat	-	-		
was	be	AUX	VBD	Mood=Ind Number=Sing Person=3 Tense=Past VerbForm=Fin			4	aux:pass	_	-
born	bear	VERB	VBN	Tense=Past VerbForm=Part Voice=Pass			0	root	-	-
in	in	ADP	IN	6	case	-	-	-	-	-
Hawaii	Hawaii	PROPN	NNP	Number=Sing	4	obl	-	SpaceAfter=No		
.	.	PUNCT	.	4	punct	-	-	-	-	-
He	he	PRON	PRP	Case=Nom Gender=Masc Number=Sing Person=3 PronType=Prs			3	nsubj:pass	-	-
was	be	AUX	VBD	Mood=Ind Number=Sing Person=3 Tense=Past VerbForm=Fin			3	aux:pass	_	-
elected	elect	VERB	VBN	Tense=Past VerbForm=Part Voice=Pass			0	root	-	-
president	president	PROPN	NNP	Number=Sing	3	xcomp	-	-	-	-
in	in	ADP	IN	6	case	-	-	-	-	-
2008	2008	NUM	CD	NumType=Card	3	obl	-	SpaceAfter=No		
.	.	PUNCT	.	3	punct	-	-	-	-	-

ბარაკ	ბარაკ	PROPN	NNP	Number=Sing	4	nsubj:pass	-	-		
ობამა	ობამა	PROPN	NNP	Number=Sing	1	flat	-	-		
დაიბადა	დაბადება	VERB	VBN	Tense=Past VerbForm=Part Voice=Pass			0	root	-	-
ჰავაიზე	ჰავაი	PROPN	NNP	Number=Sing	4	obj	-	SpaceAfter=No		
.	.	PUNCT	.	4	punct	-	-	-	-	-
ის	ის	PRON	PRP	Case=Nom Gender=Masc Number=Sing Person=3 PronType=Prs			3	nsubj:pass	-	-
აირჩიეს	არჩევა	VERB	VBN	Tense=Past VerbForm=Part Voice=Pass			0	root	-	-
პრეზიდენტად	პრეზიდენტი	PROPN	NNP	Number=Sing	3	obj	-	-	-	-
.	.	PUNCT	.	3	punct	-	-	-	-	-