



# CUNI x-ling: Parsing under-resourced languages in CoNLL 2018 UD Shared Task

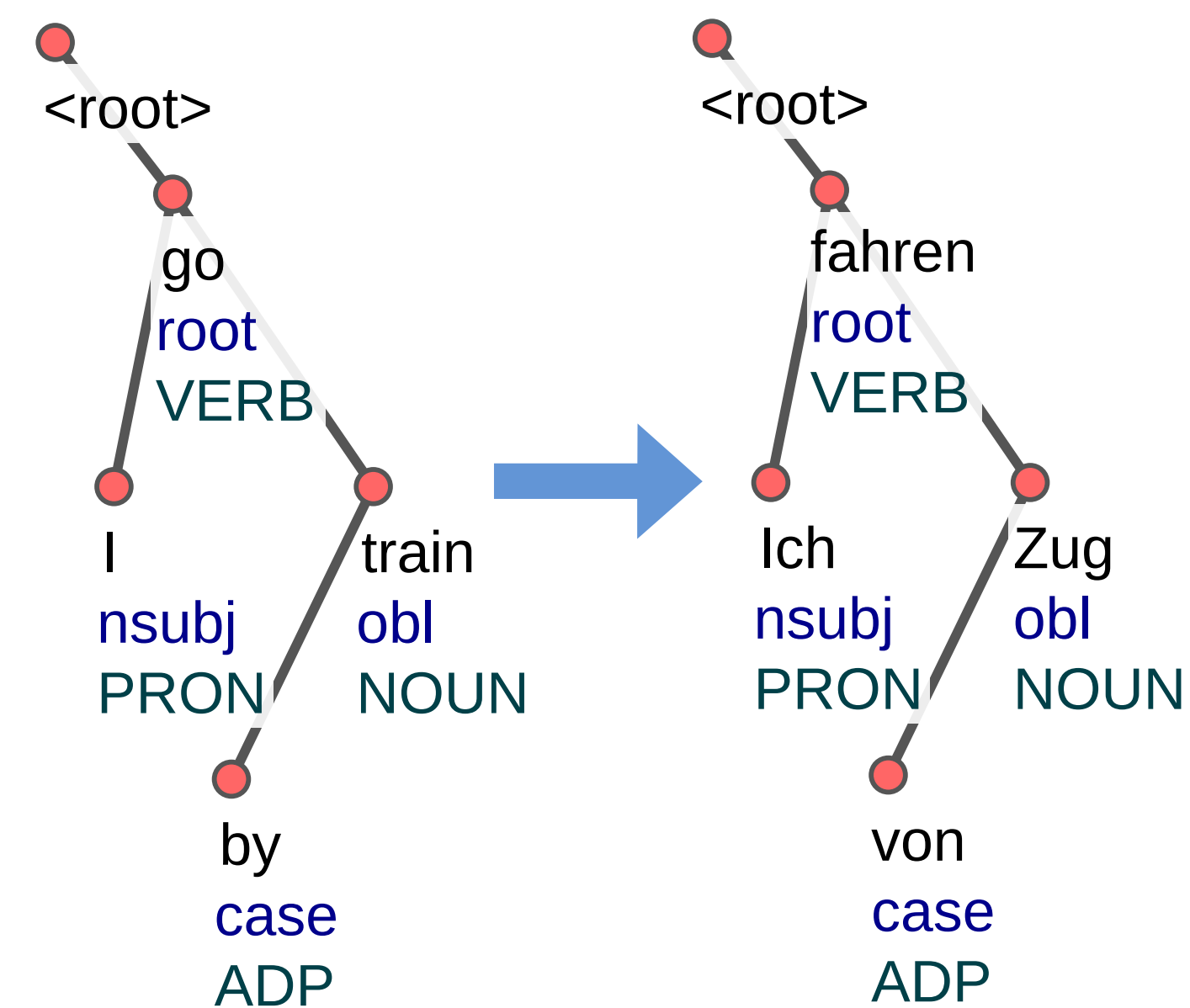
Rudolf Rosa and David Mareček

Institute of Formal and Applied Linguistics, Charles Univeristy, Prague, Czech Republic



## Treebank translation

word-alignment (FastAlign) on OpenSubtitles2018 (Opus) each word is translated by the target that was most frequently aligned to it



## Pretrained embeddings

When the UDPipe parser is trained, we use pretrained word-embeddings by Bojanovski et al. (2016) on Wikiperdia texts

## UniMorph morphology

Universal morphology annotation (Sylak-Glassman, 2016)

different features form UD mapping table e.g.

1 -> Preson=1

SG -> Number=Sg

COND -> Mood=Cnd

ACC -> Cas=Acc

PRF -> Aspect=Perf

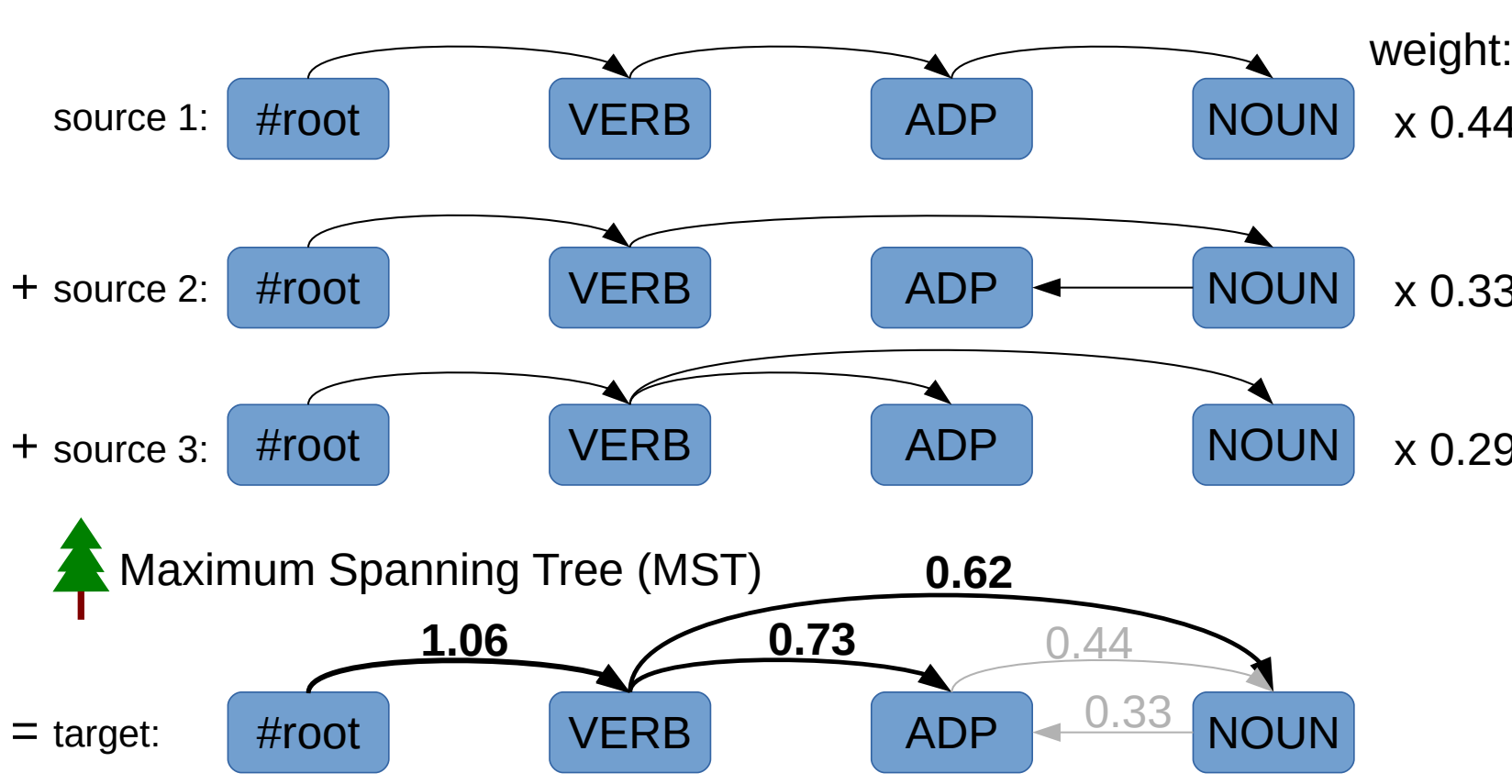
ADJ -> POS = ADJ

post-correction

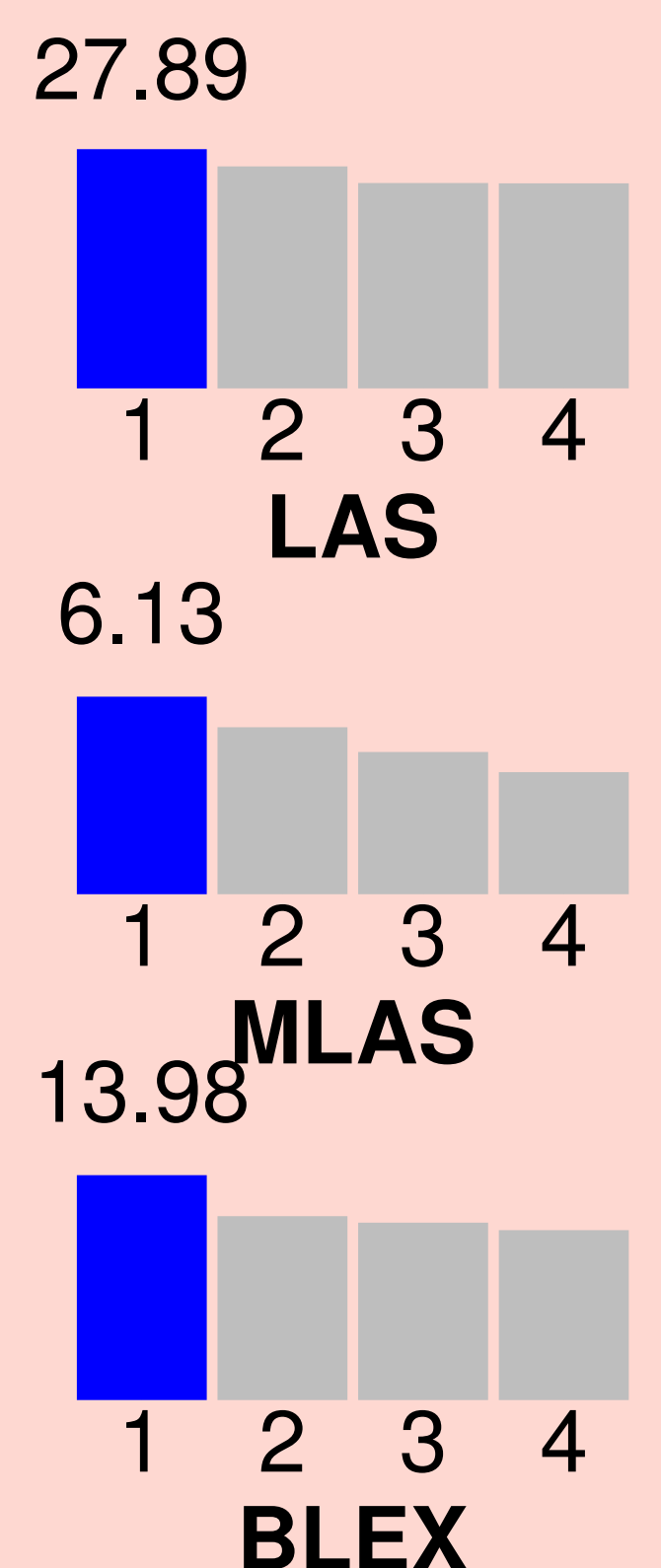
if the word is found in unimorph, chnange its tag, lemma and features

## Combining multiple parsers

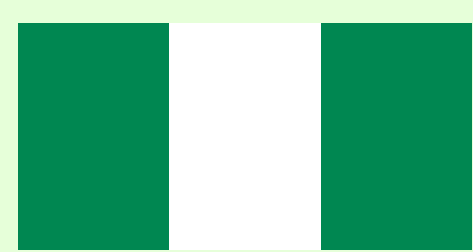
Several different weighted parse trees combined together run maximum spanning tree



## Overall results on low-resource languages



## Languages with no training data



### Naija

1. apply English tokenizer
2. "translate" words to English
3. apply English tagger and parser
4. copy form to lemma, remove final -s



#### Wiki:

A bai shu giv mai broda  
[I bought shoes that I gave to my brother]

#### Bible:

Jisos Kraist wey dem born for David and Abraham famili, na em stori bi dis.

#### jw.org:

We come from different different place and we dey speak different different language.

### Naija -> English:

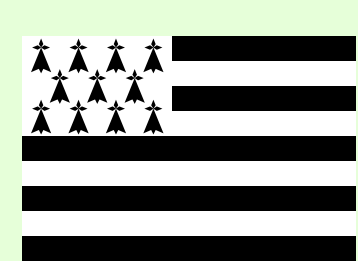
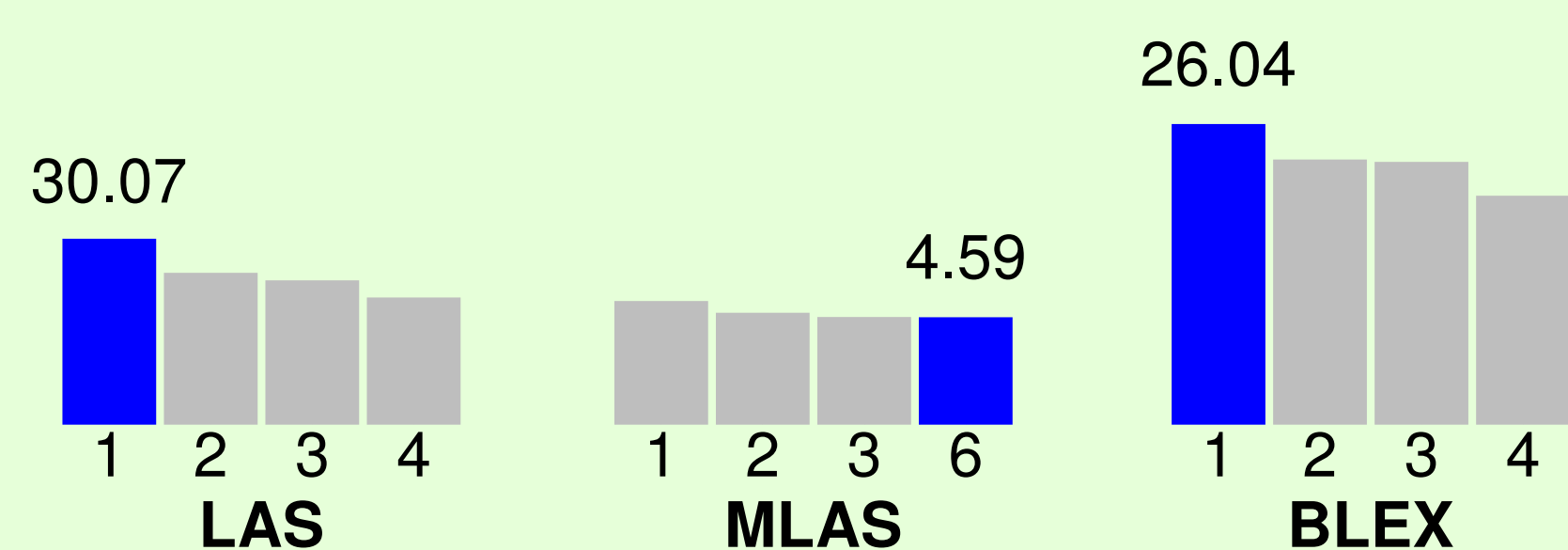
- if the Naija word is not in English lexicon:

- word changes:

|       |          |       |          |
|-------|----------|-------|----------|
| sey   | -> that  | de    | -> is    |
| na    | -> is    | don   | -> has   |
| wey   | -> which | am    | -> him   |
| im    | -> his   | go    | -> will  |
| wetin | -> what  | no    | -> not   |
| dey   | -> is    | di    | -> the   |
| deh   | -> is    | pikin | -> small |
| foh   | -> in    | sebi  | -> right |
| e     | -> he    | abi   | -> right |
| dem   | -> they  | nna   | -> man   |
| dis   | -> this  | sabi  | -> know  |

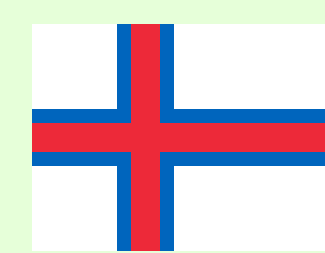
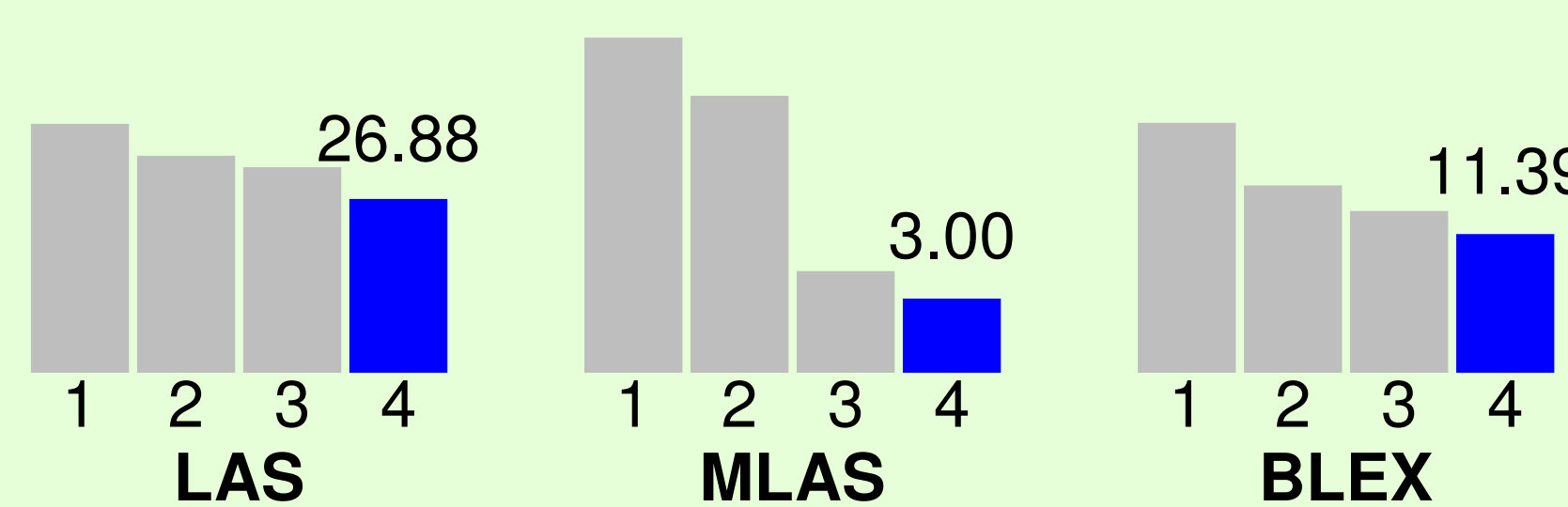
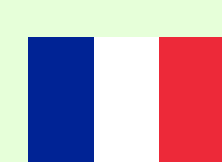
- character changes:

|     |       |    |       |
|-----|-------|----|-------|
| i   | -> y  | k  | -> c  |
| d   | -> th | >  | -> h  |
| t   | -> th | \$ | -> r  |
| a\$ | -> er | o  | -> ou |

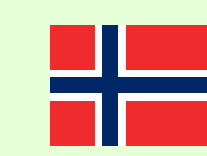


### Breton

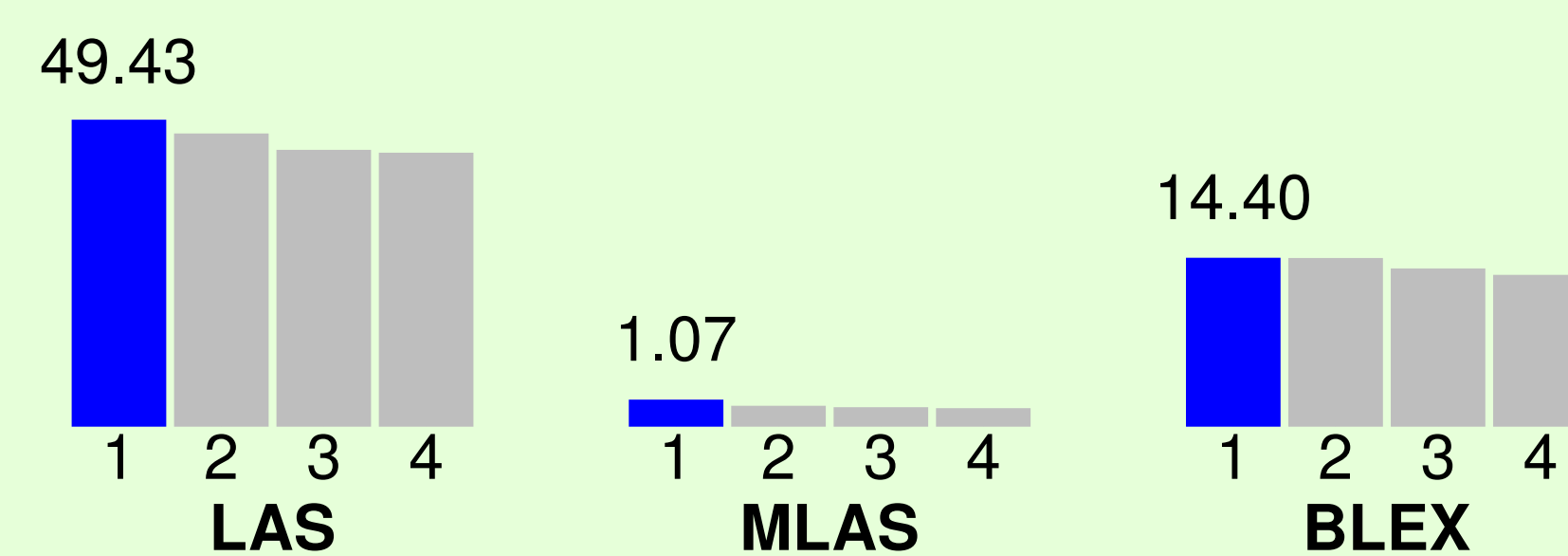
1. translate French treebank into Breton
2. train pseudo-Breton tagger and parser
3. apply French tokenizer
4. apply pseudo-Breton tagger and parser
5. apply UniMorph morphology post-correction



### Faroeese



1. train devowelled Nynorsk tagger and parser
2. apply Nynorsk tokenizer
3. apply dewovelled Nynorsk tagger and parser
4. copy lowercased form to lemma
5. apply UniMorph morphology post-correction



### Devowellization:

#### English:

Everyone has the right to life, liberty and the security of person.

#### Nynorsk:

Alle har rett til liv, fridom og personleg tryggleik.

#### Faroeese:

Ein og hvør hevur rætt til lív, frælsi og persónliga trygd.

same words: 2

**Devowelled Nynorsk:**  
Il hr rtt tl lv, frdm g prsnlg trgglk.

**Devowelled Faroeese:**  
n g hvr hvr rtt tl lv, frls g prsnlg trgd.

same words: 5



### Thai

1. obtain Thai tokenizer
2. translate indonesian, chinese and vietnamese treebanks into Thai
3. train pseudo-thai parsers on the translated treebanks
4. combine taggers and parsers based on LAS on the development data

### Tokenization:

- generate synthetic Thai text by sampling Thai tokens

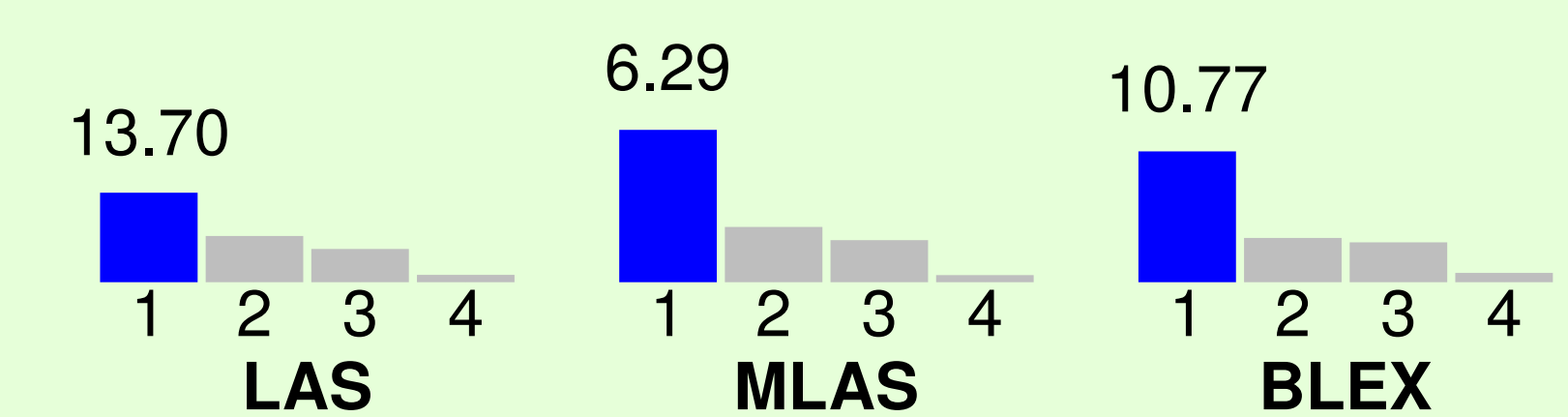
- list available at pretrained word-embeddings on Wikipedia

- token distribution:  
 $\text{Prob}(t) = 1 / \sqrt{\text{Ord}(t)}$

- English, Japanese, and Chinese filtered out

- train UDPipe tokenizer on such synthetic Thai

|  |            |      |
|--|------------|------|
|  | Indonesian | 0.75 |
|  | Chinese    | 0.55 |
|  | Vietnamese | 0.40 |



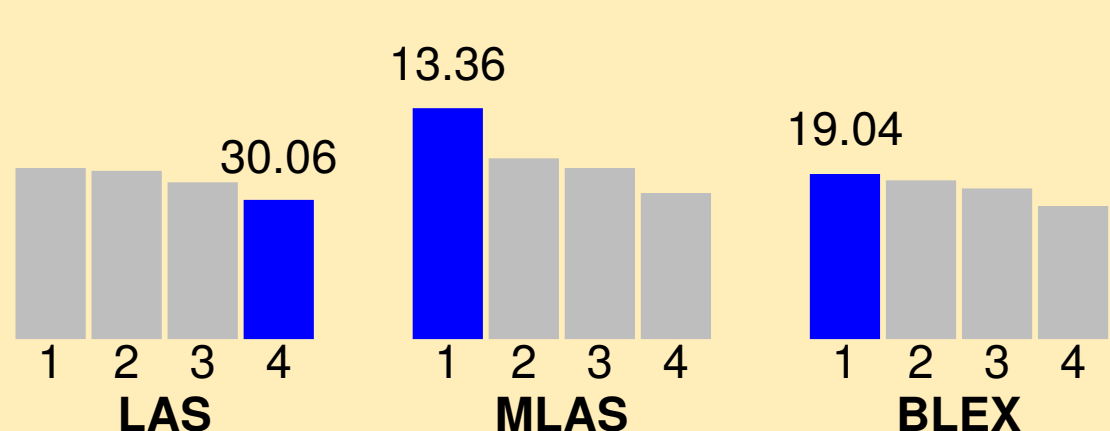
## Languages with small training data

1. train a UDPipe tokenizer, tagger, and parser
2. train delexicalized parsers for two other
3. tokenize and tag the input
4. parse it with the target parser and the other two delexicalized
5. do weighted combination of the parse trees using LAS
6. post-fix morphology using UniMorph, rewrite UPOS and lemmas



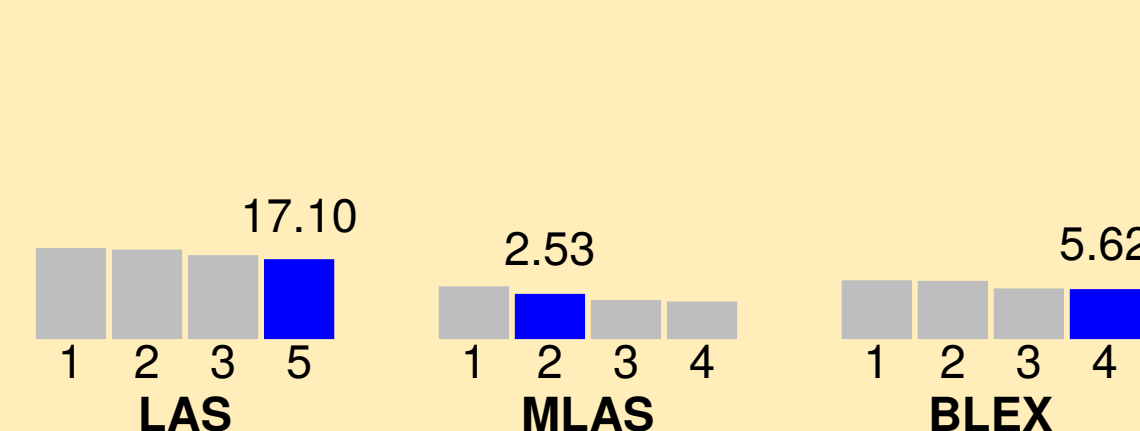
### Armenian

|  |          |      |
|--|----------|------|
|  | Armenian | 0.57 |
|  | Latvian  | 0.56 |
|  | Estonian | 0.51 |



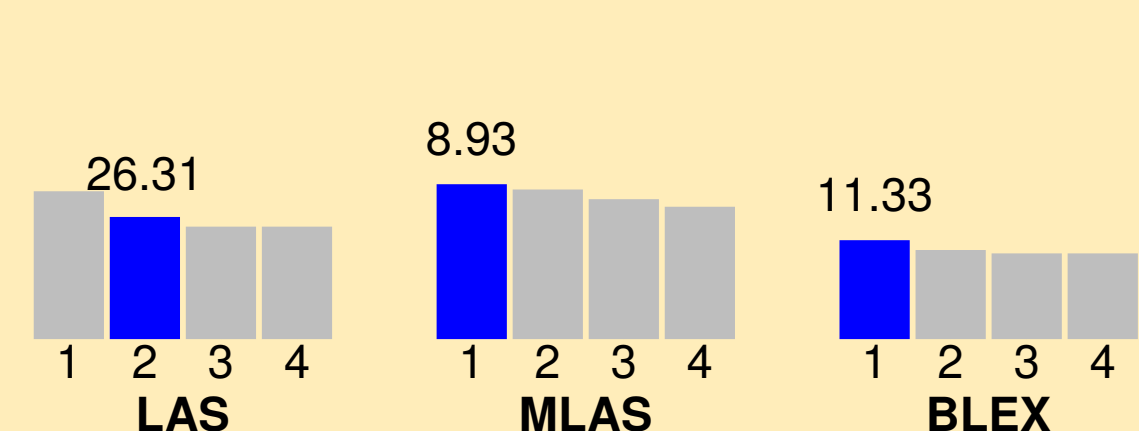
### Buryat

|  |        |      |
|--|--------|------|
|  | Buryat | 0.45 |
|  | Hindi  | 0.41 |
|  | Uyghur | 0.38 |



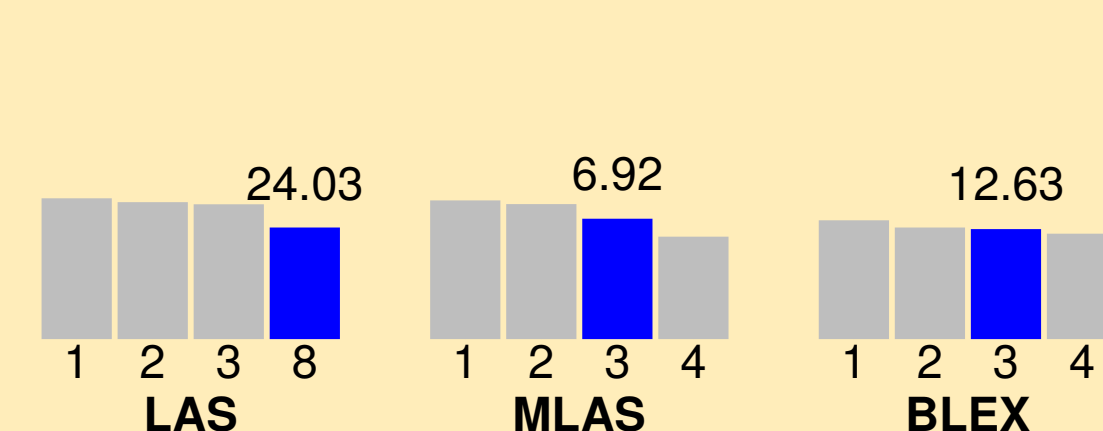
### Kazakh

|  |         |      |
|--|---------|------|
|  | Kazakh  | 0.44 |
|  | Turkish | 0.33 |
|  | Uyghur  | 0.29 |



### Kurmanji

|  |          |      |
|--|----------|------|
|  | Kurmanji | 0.52 |
|  | Latin    | 0.47 |
|  | Greek    | 0.45 |



### Up. Sorbian

|  |             |      |
|--|-------------|------|
|  | Up. Sorbian | 0.40 |
|  | Polish      | 0.60 |
|  | Czech       | 0.51 |

