Multilingual Parsing from Raw Text to Universal Dependencies

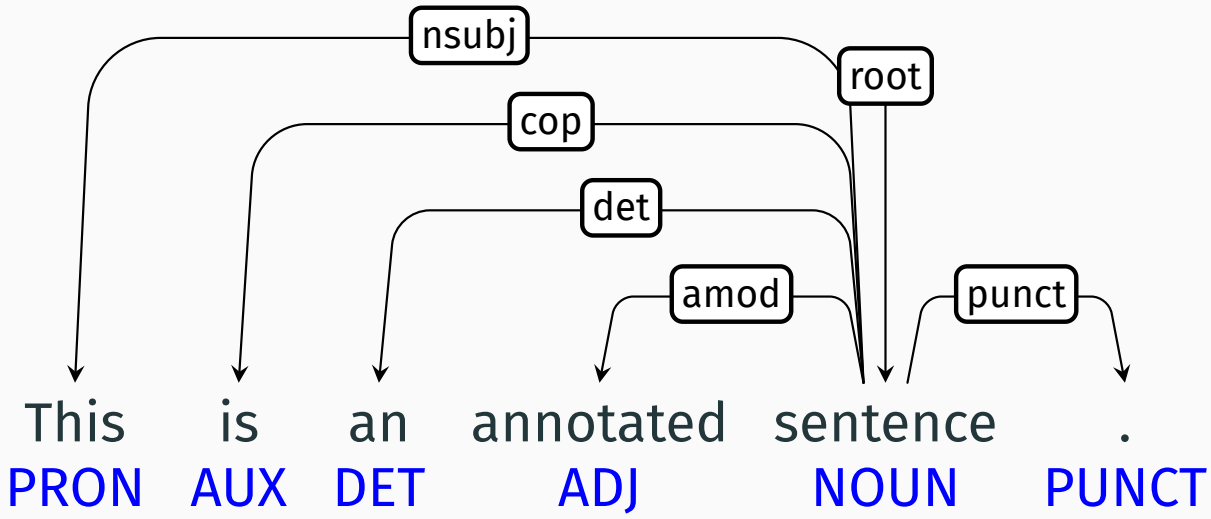
CoNLL 2017 shared task

Daniel Zeman

Institute of Formal and Applied Linguistics, Charles University

In collaboration with **Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Martin Potthast, Filip Ginter, Juhani Luotolahti, Slav Petrov** and many others

Universal Dependencies and Dependency Parsing



UD Treebanks

Ancient Greek	202K	LF
Ancient Greek-PROIEL	211K	LF
Arabic	242K	LF
Arabic-NYUAD	629K	LF
Arabic-PUD	20K	LF
Basque	121K	LF
Belarusian	6K	LF
Bulgarian	156K	LF
Buryat	10K	LF
Catalan	530K	LF
Chinese	123K	LF
Chinese-PUD	21K	F
Coptic	3K	LF
Croatian	197K	LF
Czech	1,330K	LF
Czech-CAC	493K	LF
Czech-CLTT	37K	LF
Czech-PUD	18K	LF
Danish	100K	LF
Dutch	209K	LF
Dutch-LassySmall	101K	LF
English	254K	LF
English-ESL	88K	L
English-LinES	82K	
English-PUD	21K	LF
English-ParTUT	49K	LF
Estonian	47K	LF
Finnish	202K	LF
Finnish-FTB	159K	LF
Finnish-PUD	15K	LF
French	391K	LF
French-FTB	556K	LF
French-PUD	24K	F
French-ParTUT	27K	LF
French-Sequoia	68K	LF
Galician	138K	LF
Galician-TreeGal	23K	LF

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

CoNLL 2008: + semantic dependencies (English)

CoNLL 2009: + semantic dependencies (ca, cs, de, en, es, ja, zh)

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

CoNLL 2008: + semantic dependencies (English)

CoNLL 2009: + semantic dependencies (ca, cs, de, en, es, ja, zh)

ICON 2009 (Hindi, Bangla, Telugu)

ICON 2010 (Hindi, Bangla, Telugu)

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

CoNLL 2008: + semantic dependencies (English)

CoNLL 2009: + semantic dependencies (ca, cs, de, en, es, ja, zh)

ICON 2009 (Hindi, Bangla, Telugu)

ICON 2010 (Hindi, Bangla, Telugu)

SPMRL 2013 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

SPMRL 2014 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

CoNLL 2008: + semantic dependencies (English)

CoNLL 2009: + semantic dependencies (ca, cs, de, en, es, ja, zh)

ICON 2009 (Hindi, Bangla, Telugu)

ICON 2010 (Hindi, Bangla, Telugu)

SPMRL 2013 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

SPMRL 2014 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

VarDial 2017 (cross-lingual: cs-sk, sl-hr, da/sv-no)

Dependency Parsing Shared Tasks

CoNLL 2006 (13 langs: ar, cs, bg, da, de, es, ja, nl, pt, sl, sv, tr, zh)

CoNLL 2007 (10 langs: ar, ca, cs, el, en, eu, hu, it, tr, zh)

CoNLL 2008: + semantic dependencies (English)

CoNLL 2009: + semantic dependencies (ca, cs, de, en, es, ja, zh)

ICON 2009 (Hindi, Bangla, Telugu)

ICON 2010 (Hindi, Bangla, Telugu)

SPMRL 2013 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

SPMRL 2014 (9 languages: ar, de, eu, fr, he, hu, ko, pl, sv)

VarDial 2017 (cross-lingual: cs-sk, sl-hr, da/sv-no)

CoNLL 2017 (45 languages + surprise + end-to-end parsing)

Languages and Treebanks

All UD 2.0 treebanks except:

- Too small
- Non-free

Languages and Treebanks

All UD 2.0 treebanks except:

- Too small
- Non-free

Arabic NYUAD: not available free of charge

Languages and Treebanks

All UD 2.0 treebanks except:

- Too small
- Non-free

Arabic NYUAD: not available free of charge

At least 10K test words $\Rightarrow$

- Exclude: Belarusian, Coptic, Lithuanian, Sanskrit, Tamil
- Include but small training: French ParTUT, Galician TreeGal, Irish, **Kazakh**, Latin, Slovenian SST, Ukrainian, **Uyghur**

Languages and Treebanks

All UD 2.0 treebanks except:

- Too small
- Non-free

Arabic NYUAD: not available free of charge

At least 10K test words $\Rightarrow$

- Exclude: Belarusian, Coptic, Lithuanian, Sanskrit, Tamil
- Include but small training: French ParTUT, Galician TreeGal, Irish, **Kazakh**, Latin, Slovenian SST, Ukrainian, **Uyghur**

Surprise languages

- Buryat, Kurdish, Northern Sámi, Upper Sorbian

Languages and Treebanks

All UD 2.0 treebanks except:

- Too small
- Non-free

Arabic NYUAD: not available free of charge

At least 10K test words $\Rightarrow$

- Exclude: Belarusian, Coptic, Lithuanian, Sanskrit, Tamil
- Include but small training: French ParTUT, Galician TreeGal, Irish, **Kazakh**, Latin, Slovenian SST, Ukrainian, **Uyghur**

Surprise languages

- Buryat, Kurdish, Northern Sámi, Upper Sorbian

New parallel test set (DFKI, Google and others):

- 14 languages in the task, 4 others exist

Additional Data

Just one “closed” track

Registered participants were asked for suggestions

CommonCrawl + word embeddings

Word Atlas of Language Structures (WALS)

Wikipedia Dumps

- Wikipedia word vectors (90 languages) by Facebook

Opus Parallel Corpora

WMT 2016 Parallel + Monolingual Data

Apertium + Giellatekno Morphological Analyzers

French Treebank UD v2 conversion

Multi-Language and Multi-Domain

English language

- **UD English** *(Web Treebank)*: blog, social, reviews
  - 205K train, 25K dev, 25K test
- **UD English LinES:** fiction, nonfiction (sw localization), spoken
  - 50K train, 17K dev, 16K test
- **UD English ParTUT:** legal, news, wiki
  - 26K train, 12K dev, 12K test
- **UD English PUD:** news, wiki
  - roughly 20K test only!

One model for all... but different domains!

81 test files in total

Main system score:

- macro-average LAS across all test sets (not languages)

End-to-End Parsing

A real-world scenario

No gold-standard processing available in the test data

End-to-End Parsing

A real-world scenario

No gold-standard processing available in the test data

Sentence segmentation

End-to-End Parsing

A real-world scenario

No gold-standard processing available in the test data

Sentence segmentation

Tokenization

Word segmentation (multi-word tokens)

End-to-End Parsing

A real-world scenario

No gold-standard processing available in the test data

Sentence segmentation

Tokenization

Word segmentation (multi-word tokens)

Morphological analysis

- If your parser needs it
- Exception: predicted morphology available for surprise languages

End-to-End Parsing

A real-world scenario

No gold-standard processing available in the test data

Sentence segmentation

Tokenization

Word segmentation (multi-word tokens)

Morphological analysis

- If your parser needs it
- Exception: predicted morphology available for surprise languages

Parsing

Baseline Models

UDPipe (ÚFAL): trained segmenter, tagger+lemmatizer, parser

Pre-processed test data (except syntax) directly available

Just use that if you don't have anything better

SyntaxNet / ParseySaurus (Google)

No interest in surprise languages?

Use simple delexicalized parser.

Evaluation Metrics

Align system-output tokens to gold tokens

Al-Zaman : American forces killed Shaikh Abdullah al-Ani, the preacher at the mosque in the town of Qaim, near the Syrian border.

GOLD:	Al	-	Zaman	:	American	forces	killed	Shaikh
OFFSET:	0-1	2	3-7	9	11-18	20-25	27-32	34-39

All characters except for whitespace match => easy align!

SYSTEM:	Al-Zaman	:	American	forces	killed	Shaikh
OFFSET:	0-7	9	11-18	20-25	27-32	34-39

Evaluation Metrics

Align system-output tokens to gold tokens

Die Kosten sind definitiv auch im Rahmen.

GOLD:	Die	Kosten	sind	definitiv	auch	im	Rahmen	.
SPLIT:	Die	Kosten	sind	definitiv	auch	in dem	Rahmen	.
OFFSET:	0-2	4-9	11-14	16-24	26-29	31-32	34-39	40

Corresponding but not identical spans?

Find longest common subsequence

SYSTEM:	Kosten	sind	definitiv	auch	im	Rahmen	.
SPLIT:	Kosten	sind	de finitiv	auch	im	Rahmen	.
OFFSET:	4-9	11-14	16-24	26-29	31-32	34-39	40

Evaluation Metrics

Align system-output tokens to gold tokens

Die Kosten sind definitiv auch im Rahmen.

GOLD:	Die	Kosten	sind	definitiv	auch	im	Rahmen	.
SPLIT:	Die	Kosten	sind	definitiv	auch	in dem	Rahmen	.
OFFSET:	0-2	4-9	11-14	16-24	26-29	31-32	34-39	40

Corresponding but not identical spans?

Find longest common subsequence

SYSTEM:	auch	im	Rahmen	.
SPLIT:	auch	in einem , dem alle zustimmen ,	Rahmen	.
OFFSET:	26-29	31-32	34-39	40

Evaluation Metrics

Word IDs no longer match between gold and system files!

Instead of comparing gold HEAD to system HEAD

$$head_{System}(i) = head_{Gold}(i)$$

(Comparing just integers here.)

Evaluation Metrics

Word IDs no longer match between gold and system files!

Instead of comparing gold HEAD to system HEAD

$$head_{System}(i) = head_{Gold}(i)$$

(Comparing just integers here.)

Compare aligned nodes, if alignment is found

$$node : Integer \rightarrow Node$$

$$align : SystemNode \rightarrow GoldNode$$

$$align(head_{System}(node_i)) = head_{Gold}(align(node_i))$$

(Comparing node objects.)

Evaluation Metrics

Word IDs no longer match between gold and system files!

Instead of comparing gold HEAD to system HEAD

$$head_{System}(i) = head_{Gold}(i)$$

(Comparing just integers here.)

Compare aligned nodes, if alignment is found

$$node : Integer \rightarrow Node$$

$$align : SystemNode \rightarrow GoldNode$$

$$align(head_{System}(node_i)) = head_{Gold}(align(node_i))$$

(Comparing node objects.)

Cannot align? No point for attachment!

Evaluation Metrics

Word IDs no longer match between gold and system files!

Instead of comparing gold HEAD to system HEAD

$$head_{System}(i) = head_{Gold}(i)$$

(Comparing just integers here.)

Compare aligned nodes, if alignment is found

$$node : Integer \rightarrow Node$$

$$align : SystemNode \rightarrow GoldNode$$

$$align(head_{System}(node_i)) = head_{Gold}(align(node_i))$$

(Comparing node objects.)

Cannot align? No point for attachment!

Wrong sentence boundary?

$\Rightarrow$ one or more wrong relations

Labeled Attachment Score

Correct relation ... alignment of parent equals to parent of alignment, and the universal prefix of dependency relation types match on both sides

Precision: $P = \frac{\#correctRelations}{\#systemNodes}$

Recall: $R = \frac{\#correctRelations}{\#goldNodes}$

LAS (labeled attachment F_1-score): $LAS = \frac{2PR}{P+R}$

Average over 81 test files $\Rightarrow$ main system score

Blind Evaluation on TIRA

Strong recommendation of SIGNLL (new 2015):

Teams submit software, not data

TIRA evaluation platform

`http://www.tira.io/`

Virtual machine for each team

- Configurable number of CPUs, RAM, disk space
- Currently no GPUs available
- OS: Ubuntu, Fedora or Windows
- Participants get admin access, can install anything
- ⇒ **improved reproducibility**

Blind Evaluation on TIRA

Running on test data:

- “Remote control” through web interface
- VM is “sandboxed”, detached from internet
- after the run:
  - Output files, STDOUT and STDERR archived in TIRA
  - State of VM before the run is restored (including disk)
  - Participants do not see any output
  - $\Rightarrow$ **prevents test data leakage**

Blind Evaluation on TIRA

Running on test data:

- “Remote control” through web interface
- VM is “sandboxed”, detached from internet
- after the run:
  - Output files, STDOUT and STDERR archived in TIRA
  - State of VM before the run is restored (including disk)
  - Participants do not see any output
  - $\Rightarrow$ **prevents test data leakage**

… but also makes the task extremely difficult

#ParsingTragedy

Debugging on development data (can see output)

but some files exist only in test data

#ParsingTragedy

Debugging on development data (can see output)

but some files exist only in test data

On-demand unblinding of runs by moderator

#ParsingTragedy

Debugging on development data (can see output)

but some files exist only in test data

On-demand unblinding of runs by moderator

Cannot see scores on test data

#ParsingTragedy

Debugging on development data (can see output)

but some files exist only in test data

On-demand unblinding of runs by moderator

Cannot see scores on test data

System runs for two days

but nobody knows that it is stuck in an endless loop

#ParsingTragedy

Debugging on development data (can see output)

- but some files exist only in test data

On-demand unblinding of runs by moderator

Cannot see scores on test data

System runs for two days

- but nobody knows that it is stuck in an endless loop
- or output files are not found
- we had to stitch results from multiple runs

#ParsingTragedy

Debugging on development data (can see output)

- but some files exist only in test data

On-demand unblinding of runs by moderator

Cannot see scores on test data

System runs for two days

- but nobody knows that it is stuck in an endless loop
  or output files are not found
  we had to stitch results from multiple runs

System finishes “successfully”

- but when the results are announced you find out that it picked a wrong model

Participants

111 registrations

Participants

111 registrations

56 teams got virtual machine

Participants

111 registrations

56 teams got virtual machine

38 logged in the TIRA interface (plus 2 org. accounts, and 2 extra VMs)

Participants

111 registrations

56 teams got virtual machine

38 logged in the TIRA interface (plus 2 org. accounts, and 2 extra VMs)

34 ran something (plus 1 org. account: baseline)

Participants

111 registrations

56 teams got virtual machine

38 logged in the TIRA interface (plus 2 org. accounts, and 2 extra VMs)

34 ran something (plus 1 org. account: baseline)

32 reached non-zero score on test data

Participants

111 registrations

56 teams got virtual machine

38 logged in the TIRA interface (plus 2 org. accounts, and 2 extra VMs)

34 ran something (plus 1 org. account: baseline)

32 reached non-zero score on test data

27 reached non-zero on each of the 81 files

(CoNLL 2006 had 17 participants)

(CoNLL 2007 had 23 participants)

Results: Macro LAS F1

	Team	LAS	Files
1.	Stanford (Stanford)	76.30	[OK]
2.	C2L2 (Ithaca)	75.00	[OK]
3.	IMS (Stuttgart)	74.42	[OK]
4.	HIT-SCIR (Harbin)	72.11	[OK]
5.	LATTICE (Paris)	70.93	[OK]
6.	NAIST SATO (Nara)	70.14	[OK]
7.	Koç University (İstanbul)	69.76	[OK]
8.	ÚFAL – UDPipe 1.2 (Praha)	69.52	[OK]
9.	UParse (Edinburgh)	68.87	[OK]
10.	Orange – Deskiñ (Lannion)	68.61	[OK]
11.	TurkuNLP (Turku)	68.59	[OK]
12.	darc (Tübingen)	68.41	[OK]
13.	BASELINE UDPipe 1.1 (Praha)	68.35	[OK]

Unofficial Results #ParsingTragedy

	Team	LAS	Files
1.	Stanford (Stanford)	76.30	[OK]
2.	C2L2 (Ithaca)	75.00	[OK]
3.	IMS (Stuttgart)	74.42	[OK]
4.	HIT-SCIR (Harbin)	72.11	[OK]
5.	LATTICE (Paris)	70.93	[OK]
6.	ParisNLP (Paris)	70.35	[OK]
7.	NAIST SATO (Nara)	70.14	[OK]
8.	Koç University (İstanbul)	69.76	[OK]
9.	Uppsala (Uppsala)	69.66	[OK]
10.	ÚFAL – UDPipe 1.2 (Praha)	69.52	[OK]
11.	LyS-FASTPARSE (A Coruña)	69.15	[OK]
12.	LIMSI (Paris)	68.90	[OK]
13.	UParse (Edinburgh)	68.87	[OK]
14.	RACAI (Bucureşti)	68.79	[OK]
15.	Orange – Deskiñ (Lannion)	68.63	[OK]
16.	TurkuNLP (Turku)	68.59	[OK]

Results: Word Segmentation

	Team	F_1
1.	IMS (Stuttgart)	98.81
2.	LIMSI (Paris)	98.68
3.	ÚFAL – UDPipe 1.2 (Praha)	98.63
4.	HIT-SCIR (Harbin)	98.62
5.	ParisNLP (Paris)	98.58
6.	Wanghao-ftd-SJTU (Shanghai)	98.55
	darc (Tübingen)	98.55
8.	BASELINE UDPipe 1.1 (Praha)	98.50
	C2L2 (Ithaca)	98.50
	IIT Kharagpur (Kharagpur)	98.50
	Koç University (İstanbul)	98.50
	LATTICE (Paris)	98.50
	LyS-FASTPARSE (A Coruña)	98.50
	METU (Ankara)	98.50
	MQuni (Sydney)	98.50
	NAIST SATO (Nara)	98.50

CLAS: a UD-specific Weighted Metric (Experimental)

Relations between content words are more important cross-linguistically

Attachment of function word = morphology in other languages

Weighted scoring of correct relations:

- **Weight = 1** for *root, nsubj, obj, iobj, csubj, ccomp, xcomp, obl, vocative, expl, dislocated, advcl, advmod, discourse, nmod, appos, nummod, acl, amod, conj, fixed, flat, compound, list, parataxis, orphan, goeswith, reparandum, dep*
- **Weight = 0** for *aux, case, cc, clf, cop, det, mark*
- **Weight = 0** for *punct*

Results: Macro CLAS

	Team	CLAS F_1	LAS F_1
1.	Stanford (Stanford)	72.57	76.30
2.	C2L2 (Ithaca)	70.91	75.00
3.	IMS (Stuttgart)	70.18	74.42
4.	HIT-SCIR (Harbin)	67.63	72.11
5.	LATTICE (Paris)	66.16	70.93
6.	NAIST SATO (Nara)	65.15	70.14
7.	Koç University (İstanbul)	64.61	69.76
8.	ÚFAL – UDPipe 1.2 (Praha)	64.36	69.52
9.	Orange – Deskiñ (Lannion)	64.15	68.61
10.	TurkuNLP (Turku)	63.61	68.59
11.	UParse (Edinburgh)	63.55	68.87
12.	darc (Tübingen)	63.24	68.41
13.	BASELINE UDPipe 1.1 (Praha)	63.02	68.35

Results: Surprise Languages

	Team	LAS F_1
1.	C2L2 (Ithaca)	47.54
2.	IMS (Stuttgart)	45.32
3.	HIT-SCIR (Harbin)	42.64
4.	Stanford (Stanford)	40.57
5.	ParisNLP (Paris)	39.23
6.	UParse (Edinburgh)	39.17
7.	Koç University (İstanbul)	38.81
8.	Orange – Deskiñ (Lannion)	38.72
9.	LIMSI (Paris)	37.57
10.	IIT Kharagpur (Kharagpur)	37.17
11.	BASELINE UDPipe 1.1 (Praha)	37.07

Results: Treebank Ranking by LAS

	Treebank	Max	MaxTeam	Avg	StDev
1.	ru_syntagrus	92.60	Stanford	71.64	±15.20
2.	hi	91.59	Stanford	73.41	±25.06
3.	sl	91.51	Stanford	69.70	±23.96
4.	pt_br	91.36	Stanford	72.58	±21.58
5.	ja	91.13	TRL	64.99	±23.45
6.	ca	90.70	Stanford	73.55	±21.10
7.	it	90.68	Stanford	74.06	±21.09
8.	cs_cac	90.43	Stanford	71.20	±12.07
9.	pl	90.32	Stanford	69.11	±21.59
10.	cs	90.17	Stanford	69.62	±12.34
11.	es_ancora	89.99	Stanford	72.53	±11.16
12.	no_bokmaal	89.88	Stanford	70.73	±20.97
13.	bg	89.81	Stanford	74.40	±20.46
14.	no_nynorsk	88.81	Stanford	66.81	±23.54
15.	fi_pud	88.47	Stanford	62.75	±19.28

Results: Treebank Ranking by CLAS

	Treebank	Max	MaxTeam	Avg	StDev
1.	ru_syntagrus	90.11	Stanford	67.83	±14.94
2.	sl	88.98	Stanford	65.77	±23.26
3.	cs	88.44	Stanford	66.98	±12.27
4.	cs_cac	88.31	Stanford	67.92	±11.89
5.	pl	87.94	Stanford	65.30	±20.61
6.	hi	87.92	Stanford	68.23	±24.29
7.	no_bokmaal	87.67	Stanford	67.18	±20.55
8.	pt_br	87.48	Stanford	66.36	±21.42
9.	fi_pud	86.82	Stanford	60.88	±18.25
10.	ca	86.70	Stanford	67.55	±20.36
11.	bg	86.53	Stanford	69.61	±20.13
12.	no_nynorsk	86.41	Stanford	62.92	±22.96
13.	it	86.18	Stanford	68.18	±19.79
14.	es_ancora	86.15	Stanford	66.90	±11.73
15.	nl_lassysmall	85.22	Stanford	63.61	±22.73

Thank You

http://universaldependencies.org/conll17/