Overview of the Dependency Parsing Task at EvaLatin 2026

Federica Iurescia, Marco Passarotti, Rachele Sprugnoli
Università Cattolica del Sacro Cuore
Largo Gemelli 1, 20123 Milan, Italy
federica.iurescia@unicatt.it; marco.passarotti@unicatt.it; rachele.sprugnoli@unicatt.it

Abstract

This paper presents the organization, methodology, and outcomes of the Dependency Parsing shared task held within the fourth edition of EvaLatin, a campaign dedicated to the evaluation of Natural Language Processing tools for Latin. EvaLatin aims to promote and advance research in language technologies for Latin, fostering the development of robust and linguistically informed computational approaches. The paper provides a detailed description of the data released for the shared task. It also outlines the evaluation framework and metrics adopted for assessing system performance. The results achieved by participating teams are reported and comparatively analyzed, highlighting strengths, limitations, and emerging trends in current approaches to Latin dependency parsing. Finally, the paper discusses the main challenges posed by the task and suggests directions for future research in the field.

Keywords: Latin, evaluation, dependency parsing

1. Introduction

EvaLatin 2026 is the fourth edition of the campaign devoted to the evaluation of Natural Language Processing (NLP) tools for the Latin language. As in 2020 (Sprugnoli et al., 2020), 2022 (Sprugnoli et al., 2022), and 2024 (Sprugnoli et al., 2024), EvaLatin is proposed as part of the *Workshop on Language Technologies for Historical and Ancient Languages* (LT4HALA), co-located with LREC 2026.[1] Similar to what happens in other international evaluation campaigns, participants were provided with shared test data that are made freely available for research purposes to encourage further improvement of language technologies for Latin. A shared evaluation script was also provided. Data, scorer and detailed guidelines are available in a dedicated GitHub repository.[2]

EvaLatin 2026[3] is organized around 2 tasks: Dependency Parsing and Named Entity Recognition (Boano et al., 2026). In what follows, the organization and results of the Dependency Parsing tasks are detailed.

2. Dependency Parsing

The aim of the task is to provide syntactic analysis of Latin texts following the Universal Dependencies (UD) framework (de Marneffe et al., 2021). The output submitted by the participants is a CoNLL-U file,[4] with indications of the syntactic head and of the dependency relations in the fields 7 (HEAD) and 8 (DEPREL), respectively. Other annotation levels (tokenization, sentence splitting, part-of-speech tagging and morphological features) are given as gold labels, since they are not the focus of the evaluation.

3. Data

No specific training data are released but participants are free to make use of any resource they consider useful for the task, including the Latin treebanks already available in the UD collection. In this regard, one of the challenges of this task is to understand which treebank (or combination of treebanks) is the most suitable to deal with new test data. The task aims at improving a state of the art that is not optimal. UD treebanks currently show different degrees of harmonization, and Latin is not an exception in this respect (Gamba and Zeman, 2023a); (Gamba and Zeman, 2023b).

Reflecting this situation, some texts in the dataset include punctuation, some do not, as this is the actual state of the art for Latin treebanks and corpora. Texts provided as test data are by one Classical author (Seneca the Younger) and one Medieval author (Thomas Aquinas) for a total of more than 8,000 tokens. Each author is taken as a representative of a specific text genre: Seneca the Younger for poetry, more specifically tragedy, with the *Phoenissae* (4,155 tokens), composed in the 1st century CE; Thomas Aquinas for prose, more specifically philosophical-theological treatise, with a portion of the *Summa Theologiae* (*ST*) (4,151 tokens), written

[1] https://lrec2026.info/

[2] https://github.com/CIRCSE/LT4HALA/tree/master/2026/data_and_doc

[3] EvaLatin is an initiative organized by the CIRCSE research centre at the Università Cattolica del Sacro Cuore in Milan, Italy. https://centridiricerca.unicatt.it/circse/en.html

[4] https://universaldependencies.org/format.html

Proceedings of LT4HALA 2026 @ LREC 2026, pages 211–218.

```
# sent_id = CaesBG4-A-01-607
# text = neque multum frumento sed maximam partem lacte atque pecore uiuunt multumque sunt in uenationibus
1   neque   neque   CCONJ   S   Polarity=Neg   _   _   _   LiLaflcat=i
2   multum  multum  ADV M   Degree=Pos  _   _   _   LASLAVariant=2|LiLaflcat=i
3   frumento    frumentum   NOUN    A2  Case=Abl|Gender=Neut|InflClass=IndEur0|Number=Sing  _   _   _   LiLaflcat=n2
4   sed sed CCONJ   S   _   _   _   _   LiLaflcat=i
5   maximam magnus  ADJ C1  Case=Acc|Degree=Abs|Gender=Fem|InflClass=IndEurA|Number=Sing   _   _   _   LiLaflcat=n6
6   partem  pars    NOUN    A3  Case=Acc|Gender=Fem|InflClass=IndEurI|Number=Sing   _   _   _   LiLaflcat=n3
7   lacte   lac NOUN    A3  Case=Abl|Gender=Masc|InflClass=IndEurI|Number=Sing  _   _   _   LiLaflcat=n3
8   atque   atque   CCONJ   S   _   _   _   _   LASLAVariant=1|LiLaflcat=i
9   pecore  pecus   NOUN    A3  Case=Abl|Gender=Neut|InflClass=IndEurX|Number=Sing  _   _   _   LASLAVariant=1|LiLaflcat=n3
10  uiuunt  uiuo    VERB    B3  Aspect=Imp|InflClass=LatX|Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin|Voice=Act   _   _   _   LiLaflcat=v3
11-12   multumque   _   _   _   _   _   _   _   _
11  multum  multum  ADV M   Degree=Pos  _   _   _   LASLAVariant=2|LiLaflcat=i
12  que que CCONJ   S   _   _   _   _   LiLaflcat=i
13  sunt    sum AUX B6  Aspect=Imp|InflClass=LatAnom|Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin  _   _   _   LASLAVariant=1|LiLaflcat=v6
14  in  in  ADP R   AdpType=Prep    _   _   _   LiLaflcat=i
15  uenationibus    uenatio NOUN    A3  Case=Abl|Gender=Fem|InflClass=IndEurX|Number=Plur   _   _   _   LiLaflcat=n3
```

Figure 1: Example of the test data format.

in the 13th century.[5]

Data for Seneca the Younger are taken from the *Opera Latina* corpus,[6] (Denooz, 2004) a linguistic resource manually annotated since 1961 by the Laboratoire d'Analyse Statistique des Langues Anciennes (LASLA) at the University of Liège, Belgium. Original data were converted into the annotation formalism of UD, automatically parsed with a Stanza model (Qi et al., 2020) retrained on the UD_Latin-CIRCSE treebank,[7] and manually revised by an expert of Latin language and literature.

Data for Thomas Aquinas are taken from the *Index Thomisticum* (Passarotti, 2019).[8] They were manually annotated with tokenization, sentence splitting, lemmatization, part-of-speech tagging, and syntactic annotation originally according to Prague Dependency Treebank formalism and then converted into UD format.

Data are distributed in the CoNLL-U format. Accordingly, the annotations are plain text files having the `.conllu` extension and encoded in UTF-8. An example of the test data is provided by Figure 1.

4. Evaluation

The scorer employed for the evaluation of the Dependency Parsing task is a revised version of the one developed for the *CoNLL18 Shared Task on Multilingual Parsing from Raw Text to Universal Dependencies*.[9] Our version of the script allows for the evaluation of parsing either with or without taking dependency relations subtypes into account, and is available on the EvaLatin 2026 web page.[10]

The evaluation starts by aligning the system-produced tokens to the gold standard one; given that we provide test data already tokenized, sentence-splitted and annotated with morpho-grammatical information, the alignment for tokens, sentences, words, UPOS, UFeats and lemmas should be perfect (i. e. 100.00).[11] Then, CLAS (Content-Word Labeled Attachment Score)[12] and LAS (Labeled Attachment Score)[13] are evaluated in terms of Precision, Recall, F1 both with and without subtypes.[14]

As for the baseline, we provide the scores obtained on the test data using UDPipe 2 (Straka et al., 2016) with the models trained on the UD_Latin-CIRCSE treebank v. 2.17, and the *Index Thomisticus* Treebank (UD_Latin-ITTB)[15] v. 2.17, respectively, as they are available from the tool's web interface.[16] Baseline results are given in Table 1

[5] To ensure a balanced corpus, only a portion of the *Summa Theologiae* was included in the test data.

[6] https://www.lasla.uliege.be/cms/c_8508894/fr/lasla

[7] https://github.com/UniversalDependencies/UD_Latin-CIRCSE

[8] https://itreebank.marginalia.it.

[9] https://universaldependencies.org/conll18/evaluation.html

[10] https://circse.github.io/LT4HALA/2026/EvaLatin

[11] Although UPOS values were provided in the test data, the alignment is not always perfect. On poetry data, no system achieves 100.00 on UPOS. On prose data, only THIVLVC_2 and OmnesFlores_2 do not achieve 100.00 on UPOS. Evaluation on UPOS is not performed; it is worth noting that participants experimented, among others, also on the impact of UPOS on parsing; see Section 5 and (Matsuda and Asahara, 2026); (Stymne, 2026).

[12] CLAS is the labeled F1-score over all relations except those involving function words (`aux`, `case`, `cc`, `clf`, `cop`, `det`, `mark`) and punctuation (`punct`) (Nivre and Fang, 2017).

[13] LAS is the percentage of tokens assigned both the correct `DEPREL` and `HEAD` (Buchholz and Marsi, 2006).

[14] The scorer computes also the Unlabeled Attachment Score (UAS), that is the percentage of tokens assigned the correct HEAD; the Morphology-aware Labeled Attachment Score (MLAS), that is CLAS extended with evaluation of POS tags and morphological features; the Bi-Lexical dependency score (BLEX) that combines content-word relations with lemmatization, but not with POS tags and features. These 3 metrics are not taken into account for this shared task.

[15] https://github.com/UniversalDependencies/UD_Latin-ITTB

[16] http://lindat.mff.cuni.cz/services/udpipe/

Metric	Precision	Recall	F1
CLAS	56.14	57.76	56.94
LAS	57.22	57.22	57.22

Metric	Precision	Recall	F1
CLAS	57.29	57.19	57.24
LAS	59.74	59.74	59.74

Table 1: Baseline results on poetry with subtypes on the left and without subtypes on the right.

Metric	Precision	Recall	F1
CLAS	80.33	78.51	79.41
LAS	82.17	82.17	82.17

Metric	Precision	Recall	F1
CLAS	83.00	83.14	83.07
LAS	85.21	85.21	85.21

Table 2: Baseline results on prose with subtypes on the left and without subtypes on the right.

for poetry, and 2 for prose. As the results in the Tables show, the performance is substantially higher on prose than on poetry. In addition, removing dependency subtypes consistently improves results, suggesting that fine-grained label distinctions increase task difficulty, especially in structurally complex poetic texts.

5. Participants

Four teams took part in the task. One team did not submit the report; therefore, it will not be included in this overview. Details on the participating teams and their systems are given below:

- Omnes Flores. This team submitted two runs. Their system is an extended version of an existing UD-based NLP framework that relies on multilingual Large Language Models (LLMs) with Latin data taken from the six UD_Latin treebanks. The difference in the two runs lies on how the adopted models were trained. The model used for OmnesFlores_1 has been trained with the default configuration, that takes a list of word FORM values as input. Conversely, the model used for OmnesFlores_2 has been trained taking also the values in the UPOS column of the CoNLL-U as input. Adding UPOS in the training has a positive impact on the system's performance on prose, while negatively affecting performance on poetry.
- THIVLVC. This team submitted two runs, both based on LLMs. The difference in the two runs lies in the adoption of Retrieval-Augmented Generation (RAG) on UD_Latin-CIRCSE treebank in the run THIVLVC_2, which improves system's performance both on poetry and prose.
- UppsalaNLP. This team submitted two runs, experimenting on improvement of out-of-the-box parser, via cross-lingual and cross-treebank training, with data taken from UD treebanks of historical languages. The difference in the two runs lies in using only Latin data for UppsalaNLP_2, which yields slightly lower results than UppsalaNLP_1 for prose, while performing slightly better for poetry.

6. Results and Discussion

Tables 3, 4, 5 and 6 show the final ranking for poetry (POE.) and prose (PRO.). Results are provided in terms of F1, including the baseline discussed in Section 4.

Overall, performances of all participating systems are consistently higher on prose than on poetry, confirming the tendency observed for the baseline. The best results for poetry are achieved by the THIVLVC team leveraging LLMs, particularly the run adopting RAG on UD_Latin-CIRCSE treebank (i.e., THIVLVC_2). Conversely, the same run by the THIVLVC team shows lower performances on prose. The run achieving the best results on prose relies on transfer learning and fine-tuning of historical language data (i.e., UppsalaNLP_1).

In what follows, we provide an overview of the general trends emerging from the analysis, and discuss selected results displayed in Tables 7, 8, 9, and 10. Systems achieve low performances on correctly identifying the head node of adverbial clauses (`advcl`) – ranging from roughly 25 to slightly above 45 on poetry and from nearly 50 to slightly above 70 on prose – and of adnominal clauses (`acl`) – ranging from about 23 to 60 on poetry and from slightly above 55 to nearly 75 on prose – (see Table 7).

The following example from prose data highlights the challenges involved in identifying adverbial clauses:

> **Example 1**
> Th. Aquinas *ST*. *Prooemium*
> *haec igitur et alia huiusmodi euitare studentes, tentabimus, [...] prosequi*
> Endeavoring to avoid these and other like faults, we shall try, [...] to set forth[17]

In Example 1, the token *studentes* is a present participle which is the head node of the adverbial clause in this sentence. In the gold data

[17] English translations of *Summa Theologiae* are available at: https://aquinas.cc/la/en/~ST.I.S1.

TEAM	F1 POE.
THIVLVC_2	74.03
THIVLVC_1	72.71
UppsalaNLP_2	68.47
UppsalaNLP_1	67.34
Omnes Flores_1	60.79
BASELINE	56.94
Omnes Flores_2	55.37

TEAM	F1 PRO.
UppsalaNLP_1	83.14
UppsalaNLP_2	82.17
Omnes Flores_2	81.35
THIVLVC_2	80.92
Omnes Flores_1	80.60
BASELINE	79.41
THIVLVC_1	74.04

Table 3: Dependency Parsing results in terms of CLAS with subtypes.

TEAM	F1 POE.
THIVLVC_2	76.08
THIVLVC_1	76.00
UppsalaNLP_2	69.16
UppsalaNLP_1	68.20
Omnes Flores_1	63.17
Omnes Flores_2	58.37
BASELINE	57.24

TEAM	F1 PRO.
UppsalaNLP_1	86.88
THIVLVC_2	86.60
UppsalaNLP_2	86.45
Omnes Flores_2	85.28
Omnes Flores_1	84.17
BASELINE	83.07
THIVLVC_1	81.52

Table 4: Dependency Parsing results in terms of CLAS without subtypes.

TEAM	F1 POE.
THIVLVC_2	72.88
THIVLVC_1	70.36
UppsalaNLP_2	67.56
UppsalaNLP_1	66.63
OmnesFlores_1	60.83
BASELINE	57.22
OmnesFlores_2	54.74

TEAM	F1 PRO.
UppsalaNLP_1	84.41
UppsalaNLP_2	83.74
OmnesFlores_2	83.74
THIVLVC_2	83.26
OmnesFlores_1	83.26
BASELINE	82.17
THIVLVC_1	75.72

Table 5: Dependency Parsing results in terms of LAS with subtypes.

TEAM	F1 POE.
THIVLVC_2	77.60
THIVLVC_1	76.97
UppsalaNLP_2	70.55
UppsalaNLP_1	69.59
OmnesFlores_1	65.64
OmnesFlores_2	60.20
BASELINE	59.74

TEAM	F1 PRO.
THIVLVC_2	87.93
UppsalaNLP_1	87.55
UppsalaNLP_2	87.26
OmnesFlores_2	87.09
OmnesFlores_1	86.10
BASELINE	85.21
THIVLVC_1	82.78

Table 6: Dependency Parsing results in terms of LAS without subtypes.

(see Figure 2), it is annotated as the head of an adverbial clause carrying secondary predication (`advcl:pred`). Only runs from the THIVLVC team converged with the gold annotation as `advcl:pred`. The other four runs exhibit each a different annotation: *studentes* has been annotated as

- `advcl` depending from the root identified in the token *tentabimus*;
- clausal subject (`csubj`) depending from the root identified in the token *tentabimus*;
- nominal subject (`nsubj`) depending from the token *euitare* annotated as `advcl`;
- `nsubj` depending from the root identified in the token *tentabimus*.

Such a variation exemplifies the general tendency of systems as in Table 7. Zooming in on the dependency relations occurring in annotations that diverge from gold data, indeed, we find adverbial clause, that is, the dependency relation in the gold data without the indication of the subtype. The other annotations emerging from the runs, namely nominal subject and clausal subject, are, interestingly, dependency relations for which systems exhibit overall good performances, though some challenges remain. Exploring in greater details dependency relations identifying subjects, both nominals

	acl		advcl	
	PRO.	POE.	PRO.	POE.
OmnesFlores_1	64.10	27.45	50.47	25.90
OmnesFlores_2	60.87	22.97	49.48	25.68
THIVLVC_1	56.47	39.02	64.38	46.59
THIVLVC_2	74.07	37.24	71.49	42.55
UppsalaNLP_1	70.89	60.34	52.53	40.00
UppsalaNLP_2	69.88	56.52	50.46	45.12

Table 7: F1 on selected deprel with very low scores.

	nsubj		nsubj:outer		nsubj:pass	
	PRO.	POE.	PRO.	POE.	PRO.	POE.
OmnesFlores_1	90.45	68.14	NotUsed	NotUsed	86.73	36.00
OmnesFlores_2	89.91	62.23	NotUsed	NotUsed	87.38	33.90
THIVLVC_1	90.67	84.02	NotUsed	NotUsed	84.00	59.65
THIVLVC_2	91.36	82.98	NotUsed	NotUsed	90.29	56.72
UppsalaNLP_1	92.68	77.84	0.00	0.00	92.23	65.38
UppsalaNLP_2	91.87	77.67	0.00	0.00	91.75	56.25

Table 8: F1 on the nsubj deprel and related subtypes.

	csubj		csubj:cleft		csubj:pass		csubj:relcl	
	PRO.	POE.	PRO.	POE.	PRO.	POE.	PRO.	POE.
OmnesFlores_1	85.71	23.08	NotUsed	NotUsed	89.16	NotUsed	0.00	0.00
OmnesFlores_2	85.71	35.71	NotUsed	NotUsed	89.41	NotUsed	0.00	0.00
THIVLVC_1	62.79	36.84	NotUsed	NotUsed	27.45	NotUsed	0.00	0.00
THIVLVC_2	65.85	33.33	NotUsed	NotUsed	71.43	NotUsed	0.00	13.33
UppsalaNLP_1	66.67	33.33	NotUsed	NotUsed	76.71	0.00	33.33	50.00
UppsalaNLP_2	65.79	20.00	NotUsed	0.00	72.46	NotUsed	22.22	42.11

Table 9: F1 on the csubj deprel and related subtypes.

	case		amod		det		obj	
	PRO.	POE.	PRO.	POE.	PRO.	POE.	PRO.	POE.
OmnesFlores_1	97.94	90.04	98.00	71.39	75.63	95.35	91.51	74.79
OmnesFlores_2	97.14	90.62	97.52	64.70	61.13	95.95	90.20	69.05
THIVLVC_1	96.16	97.64	89.53	78.91	69.71	74.29	95.61	89.73
THIVLVC_2	96.82	98.02	98.51	84.08	85.08	96.79	95.61	87.96
UppsalaNLP_1	96.98	92.13	96.53	78.17	77.42	96.79	94.63	76.33
UppsalaNLP_2	97.00	91.76	97.76	80.30	80.21	97.09	95.10	76.60

Table 10: F1 on selected deprel with very high scores.

(`nsubj`) and clausal (`csubj`), with their respective subtypes, the results show that they are mostly correctly identified on prose data, whereas all systems achieve lower performances for these deprels on poetry (see Tables 8 and 9, respectively).

Example 2 is instead taken from poetry data:

Example 2
Sen. *Phoen*. 214-215
turba fortunae prior abscessit a te iussa
Those who thronged around your former fortunes have left at your command[18]

[18] English translations from Seneca's *Phoenissae* are drawn by (Fitch, 2018).

As displayed in Figure 3, the token *iussa* is the head node of the adnominal clause. Only two runs from two different systems converged with the gold annotation as `acl`. In the other runs, *iussa* has been annotated differently. Three runs converge on identifying *iussa* as `advcl` depending from the root identified in the token *abscessit*. In one run *iussa* has been annotated as `advcl:pred` depending from the root identified in the token *abscessit*.

Moving away from examples towards general tendencies emerging from the data, it is worth noting that systems predict deprels or subtypes not used in gold data. This is the case, for instance, of nominal subjects of a copular clause whose predicate

Example 1
Th. Aquinas *ST*. *Prooemium*

haec igitur et alia huiusmodi euitare studentes, tentabimus, [...] prosequi
Endeavoring to avoid these and other like faults, we shall try, [...] to set forth

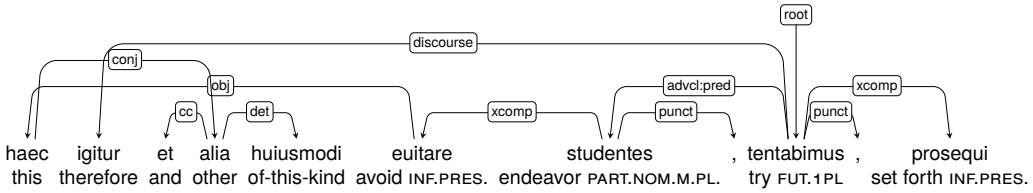

Figure 2: Syntactic tree of the first example: adverbial clause in prose text.

Example 2
Sen. *Phoen*. 214-215

turba fortunae prior abscessit a te iussa
Those who thronged around your former fortunes have left at your command

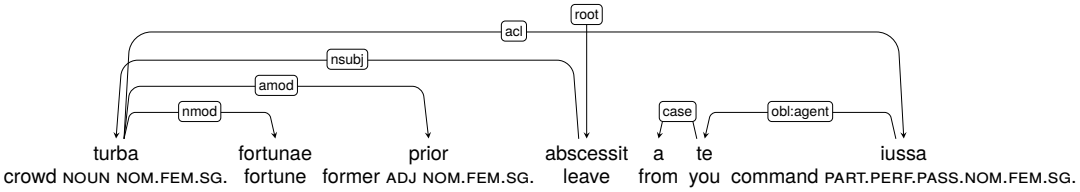

Figure 3: Syntactic tree of the second example: adnominal clause in poetry.

is itself a clause (`nsubj:outer`) in Table 8: this subtype does not occur in gold data, is predicted by UppsalaNLP systems – scoring 0 –, and is not used by other systems – appearing as NotUsed.

Conversely, some deprels occur in gold data, but are not identified by systems. This is the case, for instance, of `ccomp:reported`, used to describe reported speech, both in poetry and in prose data. Very low performances emerge particularly also with identifying `parataxis` on prose texts: the highest score is 22.22, achieved by THIVLVC_1, whereas all other systems score 0. On poetry data, results are still very low – the highest score being nearly 30 achieved by OmnesFlores_2 – but only one system exhibits 0 (OmnesFlores_1). Finally, a run from systems leveraging LLMs (i.e., THIVLVC_1) predict a deprel not used in Latin data, namely `cc:preconj`.

The best performances are achieved on deprels describing case marking (`case`) – all above 90 –, objects (`obj`) – ranging from nearly 70 for poetry to around 95 for prose –, adjectival modifiers (`amod`) – ranging from around 65 on poetry to nearly 100 on prose –, determiners (`det`) – showing a range of results slightly lower to what emerges for adjectival modifiers –, as Table 10 shows.

Overall, some deprels exhibit a moderate variation in performance (within a range of 5 percentage points): they are `nsubj`, `obj`, `amod`, `det`, `case`. A higher variation (approximately 15 percentage points) is observed for `obl`, `advmod`, and `nmod`. A consistent drop (more than 15 percentage points) emerges for `advcl` and `acl`.

7. Conclusions

The Dependency Parsing task within EvaLatin 2026 has contributed to advancing the previous state of the art: indeed, almost all partecipating systems outperform the baseline. Furthermore, in the 2024 edition, the best participant performances in the same task are lower than those achieved in the current edition. The comparison is particularly evident for poetry, as in both campaigns the test data is drawn from Senecan tragedies. More specifically, in 2024, the best system on poetry reached 74.53 F1 on CLAS without subtypes, and 75.75 F1 on LAS without subtypes (see Tables 6 and 7 in (Sprugnoli et al., 2024)). In the current edition, the best system reaches 76.08 on CLAS without subtypes, and 77.60 F1 on LAS without subtypes (see Table 4 and Table 6, respectively), with an increase of about 2 percentage points.

Nevertheless, there is still room for improvement. More specifically, as (Pommeret et al., 2026) observe, Latin UD treebanks show a degree of variation in annotation, leading to suboptimal performance of automatic models. The diversity of the available data remains an issue – that is raised for several other languages in the UD community;

currently, no strategies are able to fully address this obstacle to the optimal training of automatic systems.

From this overview, poetry emerges as more difficult than prose for parsing, as Stymne (2026) and Matsuda and Asahara (2026) also remark. A thorough investigation on the reasons for this discrepancy lies beyond the scope of this report. Future work might start exploring the impact of metrics on this aspect. In the present and in the previous edition of EvaLatin, only (Senecan) tragedy has been proposed as test data. Whether other genres within poetry do prove to be difficult for automatic parsers remains object for future research.

8. Acknowledgements

The authors thank Giovanni Moretti for customizing the scorer.

9. Bibliographical References

Valeria Boano, Eleonora Litta, and Matteo Romanello. 2026. Overview of the Named Entity Recognition Task at EvaLatin 2026. In *Proceedings of the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2026)*, Palma, Mallorca (Spain). ELRA.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, pages 149–164, New York City. Association for Computational Linguistics.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308. Retrievable at https://direct.mit.edu/coli/article/47/2/255/98516/Universal-Dependencies.

Joseph Denooz. 2004. Opera Latina : une base de données sur internet. *Euphrosyne*, 32:79–88.

J. G. Fitch. 2018. *Tragedies, Volume I: Hercules. Trojan Women. Phoenician Women. Medea. Phaedra*. Loeb classical library 62. Harvard University Press, Cambridge (MA).

Federica Gamba and Daniel Zeman. 2023a. Latin morphology through the centuries: Ensuring consistency for better language processing. In *Proceedings of the Ancient Language Processing Workshop*, pages 59–67, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Federica Gamba and Daniel Zeman. 2023b. Universalising Latin Universal Dependencies: a harmonisation of Latin treebanks in UD. In *Proceedings of the Sixth Workshop on Universal Dependencies (UDW, GURT/SyntaxFest 2023)*, pages 7–16, Washington, D.C. Association for Computational Linguistics.

Hiroshi Matsuda and Masayuki Asahara. 2026. Extending omnes flores for the EvaLatin 2026 Dependency Parsing Tasks. In *Proceedings of the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2026)*, Palma, Mallorca (Spain). ELRA.

Joakim Nivre and Chiao-Ting Fang. 2017. Universal Dependency evaluation. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 86–95, Gothenburg, Sweden. Association for Computational Linguistics.

Marco Passarotti. 2019. The project of the index thomisticus treebank. In Monica Berti, editor, *Digital Classical Philology*, pages 299–320. De Gruyter Saur, Berlin, Boston.

Luc Pommeret, Thibault Wagret, and Jules Deret. 2026. THIVLVC: Retrieval Augmented Dependency Parsing for Latin. In *Proceedings of the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2026)*, Palma, Mallorca (Spain). ELRA.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Rachele Sprugnoli, Federica Iurescia, and Marco Passarotti. 2024. Overview of the EvaLatin 2024 evaluation campaign. In *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, pages 190–197, Torino, Italia. ELRA and ICCL.

Rachele Sprugnoli, Marco Passarotti, Flavio Massimiliano Cecchini, Margherita Fantoli, and Giovanni Moretti. 2022. Overview of the EvaLatin 2022 evaluation campaign. In *Proceedings of the Second Workshop on Language Technologies for Historical and Ancient Languages*, pages 183–188, Marseille, France. European Language Resources Association.

Rachele Sprugnoli, Marco Passarotti, Flavio Massimiliano Cecchini, and Matteo Pellegrini. 2020. Overview of the EvaLatin 2020 evaluation campaign. In *Proceedings of LT4HALA 2020 -*

1st Workshop on Language Technologies for Historical and Ancient Languages, pages 105–110, Marseille, France. European Language Resources Association (ELRA).

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA). Retrievable at https://aclanthology.org/L16-1680.

Sara Stymne. 2026. UppsalaNLP at EvaLatin 2026: Multilingual parsing for Latin. In *Proceedings of the Fourth Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA 2026)*, Palma, Mallorca (Spain). ELRA.