PāliUD: Prompt-engineered annotation and philological evaluation for the Pāli Treebank

Chia-Wei Lin*
Université de Lausanne
chia-wei.lin@unil.ch

Jasvinder Singh*
Universität Leipzig
jasvinder2604@gmail.com

Khemarato Bhikkhu*
Independent Scholar
khbh@buddhistuniversity.net

Abstract

While computational research into Sanskrit has advanced significantly, closely related Middle Indo-Aryan languages like Pāli and Prakrit remain under-resourced. In our paper, we introduce our ongoing UD treebank project for Pāli and also highlight the limitations of Sanskrit-based UD parsers on our corpus. We provide annotation guidelines that are tailored to Pāli morphosyntax and document the systematic linguistic divergences from existing UD corpus of Vedic Sanskrit. Furthermore, we evaluate the efficacy of an expert linguist-in-the-loop annotation pipeline leveraging Large Language Models (LLMs). Through zero-shot and few-shot experiments, we demonstrate how LLMs can substantially speed up the creation of a high-quality treebank for a Middle Indo-Aryan language like Pāli.

1 Introduction

The Indo-Aryan languages form a major branch of the Indo-European family. They are traditionally classified into three periods: (1) Old Indo-Aryan (OIA) languages like Vedic and Classical Sanskrit, (2) Middle Indo-Aryan (MIA) languages like Prakrit and Pāli, and (3) New Indo-Aryan (NIA) languages like Hindi and Punjabi.

Pāli is used primarily as the liturgical language of Theravāda Buddhism, which has exerted significant influence on the languages and cultures of Sri Lanka and Southeast Asian countries such as Thailand, Myanmar, and Cambodia. The Pāli *Tipiṭaka*, composed in Pāli, forms the doctrinal foundation of Theravāda Buddhism. Historically, Pāli is a descendant of an OIA language closely related to Vedic Sanskrit and is considered a *koine* of various MIA dialects.

In the field of Natural Language Processing (NLP), the focus has mainly been dominated by the Sanskrit-based computational tools and resources. Despite its historical and liturgical importance, Pāli remains under-represented in the development of Indo-Aryan NLP tools.

In this paper, we introduce a treebank of the *Mahāpadānasutta* (the fourteenth *sutta* of the *Dīgha Nikāya* in the Pāli *Tipiṭaka*). We present our annotation guidelines and discuss the challenges we faced, providing a framework to help standardize future Pāli NLP research. We also conduct experiments with a less resource intensive machine-learning baseline by evaluating a Sanskrit-based UD annotation tool (UDPipe) on our Pāli dataset to highlight its limitation. Consequently, to overcome its limitations, we employ a prompt-engineered workflow to generate initial annotations for the UD treebank, which are subsequently refined through a linguist-in-the-loop process. The Pāli treebank can also be used in downstream NLP tasks such as linguistics-informed machine translation (cf. Sennrich and Haddow (2016)).

2 Related Works

There has not yet been a published Universal Dependencies Treebank corpus for Pāli. Similar work has been done for Aśokan Prakrit (Farris and Arora, 2021), Vedic Sanskrit (Hellwig et al., 2020)[1] and Classical Sanskrit.[2] More broadly, the PROIEL project (Eckhoff et al., 2018) built a large corpus of annotated treebanks for pre-modern Indo-European languages, including Latin, Ancient Greek, Gothic, Classical Armenian, Old Church Slavonic, Old Russian, Old English, Old French, Old Spanish, and Old Portuguese.

Recent Pāli NLP work has focused primarily on segmentation, corpus construction, and machine translation rather than syntactic annotation.

*Equal contribution.

[1] https://github.com/UniversalDependencies/UD_Sanskrit-Vedic

[2] https://github.com/UniversalDependencies/UD_Sanskrit-UFAL/tree/master; see also Kulkarni et al. (2020).

Tammanam et al. (2021) developed a hybrid BiLSTM and rule-based approach to Pāli sandhi segmentation. Zigmond (2023) applied unsupervised clustering to distinguish canonical from commentarial Pāli texts. Nehrdich and Keutzer (2026) introduced the MITRA parallel corpus and multilingual pretrained models for Pāli and other Buddhist classical languages. Tsukagoshi and Ohmukai (2025) showed that integrating philological commentary into model fine-tuning can improve classical-language machine translation, including in a setting using a model trained on Pāli-related Buddhist textual data. Together, these studies demonstrate growing computational support for Pāli, but none provides a machine-readable corpus of Pali grammatical annotation.

3 Treebank[3]

3.1 Text selection

Pāli is primarily studied as the scriptural language of Theravāda Buddhism. Within the Pāli Canon, the most-studied texts are in the *Sutta Piṭaka* 'Basket of the Discourses'. These books contain some of the earliest Pāli texts (Wynne, 2006), and their most prototypical form (Bodhesako, 1987; von Hinüber, 1996). For our work, we selected the *Mahāpadānasutta* (DN 14) from the middle of the first collection. A "long" (*dīgha*) discourse in mixed prose and verse with a number of repetitive sections, the text contains the major elements and typical styles of a Pāli *sutta*, making it an excellent exemplary text for commencing Pāli UD annotations.

So far, our UD corpus contains 1764 tokens, but we plan on continuously expanding the corpus.

3.2 Annotation guidelines for Pali

Our annotation guidelines are primarily based on the Vedic Sanskrit Treebank by Hellwig et al. (2020). However, to better accommodate the UD infrastructure to the linguistic structure of Pāli, we propose some modifications for the guidelines tailored to Pāli.

3.2.1 Tokenization

Like Sanskrit, Pāli employs *sandhi*: a set of phonological rules governing the merging of adjacent words across token boundaries. Unlike Sanskrit, where *sandhi* rules are highly complex, Pāli *sandhi* consists primarily of vowel elisions. Whereas Hellwig et al. (2020) resolve *sandhi* as a pre-processing step, we retain the traditional orthography without segmenting *sandhi* junctures. To mark tokens involved in *sandhi*, we use `SpaceAfter=No` in the `MISC` column. For instance:

```
# text = idaṃpissa
1   idaṃ idaṃ SpaceAfter=No
2   pi  pi  SpaceAfter=No
3   ssa idaṃ     _
```

Following Hellwig et al. (2020), we strip out all punctuation marks from our text, as these are not traditionally part of Pāli orthography.

3.2.2 Lemma

Since many Pāli nouns display multiple stem variants and consonantal nominal stems are often simplified to thematic stems, we decided to use the etymological stem as lemma when they are attested in the Pāli corpus. For example, the `LEMMA` of *rājā* 'kind' (nom. sg. masc.) is *rājan* with the etymological n-stem; the `LEMMA` of *attā* 'self' (nom. sg. masc.) is *attan*; the `LEMMA` of *ti* is *iti*.

For verbs, following the convention of Pāli scholarship, we use the 3SG. PRES. IND. form of the verb instead of the verbal root as in Hellwig et al. (2020). For example, in the Vedic Sanskrit treebank, the `lemma` of *kṛtvā* 'having done' (CONVERB) is *kṛ* (ROOT). In our Pāli treebank, the `lemma` of *katva* 'having done' (CONVERB) is *karoti* (3SG. PRES. IND.). Using 3SG. PRES. IND. resolves the problems where (1) the etymological root of a Pāli verb may not be immediately obvious to annotators without background in Indo-Aryan historical linguistics (e.g. Pāli *passati* 'to see' from the suppletive root *dṛś*, *suṇāti* 'to listen' from the root *śru*) and (2) some derived verbal stems display Pāli innovation that cannot be traced back to Sanskrit (e.g. Pāli *deti* 'to give' from the root *dā*, cf. Sanskrit *dadāti*; Pāli *paheti* 'to send' from the root *hi*, cf. Sanskrit *prahiṇoti*).

3.2.3 UPOS and Features

As in the Vedic Sanskrit treebank, we annotate all the verbal forms, including participles, as `VERB`, even in cases where the usage of the participles is adjacent to `ADJ`. This is to avoid forcing a decision on the syntactic ambiguity of Pāli participles, which can either be used as paratectic construction to the finite verb or as attribute to a noun.

[3]The treebank data is available at https://github.com/UniversalDependencies/UD_Pali-PaliCanon/tree/dev

For negated participles and adjectives with the privative *a-*, we add `Polarity=Neg` in `FEATS`. For example, *amakkhito* 'unsoiled' has the following `FEATS`:

```
Case=Nom|Gender=Masc|Number=Sing|Polarity=
Neg|Tense=Past|VerbForm=Part|Voice=Pass
```

This avoids the necessity of creating an independent lemma for each negated adjective and participles.

3.2.4 Dependency Relations

For `DEPREL`, we generally follow the guidelines of the Vedic treebank. For compounds, we use Multi-Word Token ranges to split them into their component parts. In keeping with traditional Indic grammatical analysis of compounds, we add seven new, language-specific `compound` subtypes for the compound types listed in Table 1.

DEPREL	Description
`compound:dv`	*dvandva*: compounds that express a coordinative relation[3]
`compound:tp`	*tatpuruṣa*: compounds whose first member is in an oblique relationship with the second member
`compound:dg`	*dvigu*: compounds whose first member is a numeral
`compound:kd`	*karmadhāraya*: compounds whose first member qualifies the second member[4]
`compound:na`	*nañtatpuruṣa*: compounds whose first member is a negation (*a-*) [5]
`compound:av`	*avyayībhāva*: adverbial compounds whose first member is an indeclinable element
`compound:bv`	*bahuvrīhi*: exocentric compounds indicating possessive relationship[6]

Table 1: Dependency relations for Pāli compounds

[3] `compound:coord` in Hellwig et al. (2020).

[4] `compound:name` in Hellwig et al. (2020).

[5] Although *dvigu*, *karmadhāraya*, *nañtatpuruṣa* are traditionally classified as special subtypes of *tatpuruṣa* according to Sanskrit grammarians, we make them independent classes to render the relationship more evident.

[6] Since *bahuvrīhi* is an exocentric compound, its dependency can be expressed by `amod` in the UD framework. We use `compound:bv` mainly for compounds with prepositions such as *sa-* 'with', *nir* 'without', for instance, *sadevaka* 'with gods', *nippurisa* 'without men'.

LAS	UAS	LOS
0.741 [0.756]	0.801 [0.829]	0.839 [0.854]

Table 2: Inter-annotator agreement (IAA) based on Cohen's κ and proportional agreement (in square brackets) for the labeled attachment score (LAS), unlabeled attachment score (UAS) and label-only score (LOS).

We believe that these compound subtypes will be both easier for annotators familiar with traditional grammars and will provide more precise data for statistical research of Pāli compounds.

We do not use any of Hellwig et al. (2020)'s `advcl` subtypes: `advcl:advers`, `advcl:caus`, `advcl:ccomp`, `advcl:concess`, `advcl:cond`, `advcl:consec`, `advcl:dpct`, `advcl:fin`, `advcl:lcl`, `advcl:manner`, or `advcl:tcl`. We believe that keeping only `advcl` will reduce subjective bias and inter-annotator disagreement, since the semantics of converbs and adverbial clauses are often polyvalent in Pāli.

We also do not use various `obl` subtypes in the Vedic treebank: `obl:lmod`, `obl:path`, `obl:tmod`, `obl:instr`, `obl:source`, `obl:manner`, `obl:soc`, `obl:grad`, or `obl:benef`, except we use `obl:goal` to disambiguate the accusative of goal in connection with verbs of motion (e.g. *gacchati* to go', *kamati* 'to march') from the accusative object (`obj`) of transitive verbs. For the other subtypes, their specific functions can be inferred from the `NOUN`'s `Case` in the `FEATS` column. We use only `obl` to avoid inter-annotator disagreement due to subjective interpretation of the semantics of cases. For instance, `Case=Loc` can signify either a temporal (`obl:tmod`) or a locative (`obl:lmod`) relation and `Case=Ins` can signify either an instrumental (`obl:instr`) or a sociative (`obl:soc`) relation.

3.3 LLM-assisted workflow

To produce the draft annotation, we used the Vedic Sanskrit Treebank by Hellwig et al. (2020) as in-context examples and prompted the LLM (Gemini-3.1-Pro) to generate a draft annotation of the Pāli text in CoNLL-U format. We then manually corrected the output. For subsequent batches, we replaced the Vedic examples with our own corrected Pāli annotations, included our annotation guidelines in the prompt, and iteratively prompted the LLM to produce the next batch of annotation. A similar approach has recently been adopted by Lin and Luinetti (2026) to establish a UD treebank for Old

Statistic	Value
Total Sentences	158
Total Tokens	1,764
Avg. Sentence Length	11.16
UPOS Tag	**Count**	**%**
NOUN	780	44.22
VERB	217	12.30
ADJ	163	9.24
ADV	145	8.22
PRON	93	5.27
PART	117	6.63
PROPN	93	5.27
NUM	70	3.97
CCONJ	25	1.42
AUX	21	1.19
Others (ADP, SCONJ)	5	0.28
Top Deprels	**Count**
appos	198
nsubj	190
advmod	159
root	158
amod	122

Table 3: Descriptive statistics, UPOS distribution, and top 5 dependency relations for the manually-annotated part of Pāli treebank

Georgian.

4 Experimental setup

We evaluated the existing tools of UD treebank with Pāli annotation. We test the Sanskrit-based UD annotation tools, provided by UDPipe 2 (Straka, 2018) on our corpus. Furthermore, we evaluate the LLM-assisted annotation with our linguistic and philological knowledge. Our pipeline is illustrated in Figure 1.

4.1 UDPipe

After applying the Sanskrit UDPipe 2 model to our Pāli source text, we obtained the evaluation results as shown in Table 4. These results indicate a substantial domain mismatch between the two languages. The model also produces fewer tokens than the gold standard (1,377 predicted vs. 1,764 gold), suggesting that the sandhi- and compound-boundary rules learned from Sanskrit do not transfer well to Pāli. This performance drop-off underscores the impact of linguistic divergences between OIA and MIA languages as the model's internal 'Sanskritized' embeddings could not recognize the lexemes in Pāli.

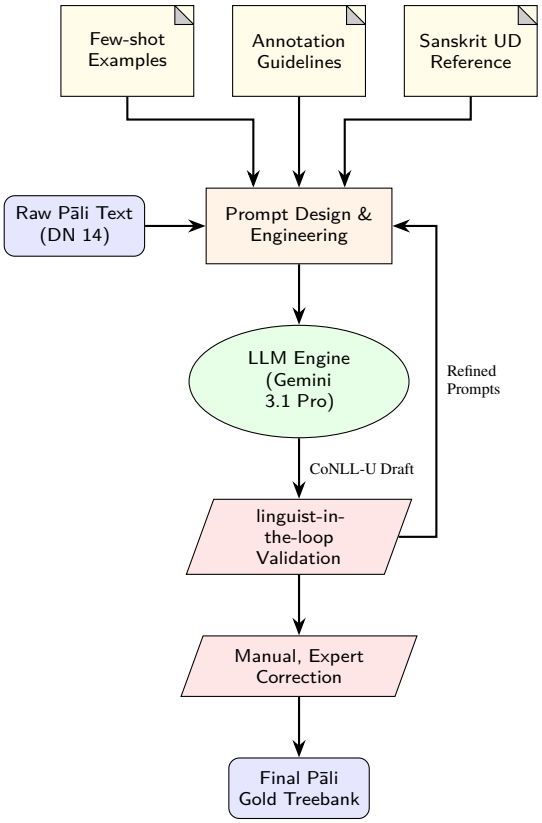

Figure 1: Overview of the linguist-in-the-loop annotation pipeline for developing the Pāli Universal Dependencies treebank.

Model	Precision	Recall	F1-score
Sanskrit UDPipe 2	23.0	24.0	23.5
Zero-shot Gemini	32.0	28.4	30.0
Few-shot Gemini	**85.3**	**91.5**	**88.2**

Table 4: Comparison of macro-averaged results (%) between Vedic Sanskrit UDPipe and LLM (Gemini) prompting methods for UD annotation.

4.2 Evaluation of LLM-generated annotations

We evaluate the LLM's performance in the annotation task by benchmarking zero-shot against few-shot prompting. In the few-shot setting, the model is prompted with human-annotated passages from the *Mahāpadānasutta* and a set of instructions derived from our revised annotation guidelines. Evaluation was performed on a test data of four paragraphs, with performance metrics reported in Table 4.

In the initial zero-shot output of the LLM, we noticed several imprecisions in the annotations. For instance, the accusative of goal linked to verbs of motion is often marked `obj` instead of `obl:goal` in `DEPREL`. Moreover, `LEMMA` was often provided by the LLM in their Sanskrit forms, which we had to manually correct to their Pāli forms. This sug-

gests that LLMs' understanding of Pāli is heavily influenced by its Sanskrit training data, which is more abundant than Pāli and reinforces the need for additional, high-quality training data for Pāli specifically.

5 Future work

In the future, we plan to expand our corpus to include other *sutta*s in the *Dīgha Nikāya* and other parts of the Pāli canon. The annotated corpus will also enable quantitative corpus-linguistic research on the Pāli corpus, for example, to study the statistical distribution of variants of Pāli nominal and verbal forms in order to better understand Pāli textual and linguistic history.

Limitations

As this work originated as a small group project, resource constraints limited the range of Large Language Models (LLMs) available for the prompt-engineering experiments. The scope of our corpus is subject to the limitation of the small number of annotators with sufficient knowledge of Pāli.

Acknowledgments

We thank Dr. Diego Luinetti for his insights into LLM-assisted workflows for treebank annotation and Dibyajoti Jana for his discussion of the guidelines for Pāli. We thank Prof. Sebastian Nehrdich for inspiring us to create a treebank of Pāli. We thank Dr. Mudagamuwe Maithrimurthi, Prof. Dr. Ingo Strauch and Prof. Dr. Jowita Kramer for their generous support and guidance. We would also like to thank tipitaka.app for curating and making the Pāli canon data freely available online.

References

Sāmanera Bodhesako. 1987. *Beginnings: The Pali Suttas*. Number 313 in Wheel. Buddhist Publication Society, Kandy, Sri Lanka.

Hanne Eckhoff, Kristin Bech, Gerlof Bouma, Kristine Eide, Dag Haug, Odd Einar Haugen, and Marius Jøhndal. 2018. The PROIEL treebank family: a standard for early attestations of Indo-European languages. *Language Resources and Evaluation*, 52(1):29–65.

Adam Farris and Aryaman Arora. 2021. For the Purpose of Curry: A UD Treebank for Ashokan Prakrit. In *Proceedings of the Fifth Workshop on Universal Dependencies (UDW, SyntaxFest 2021)*. Association for Computational Linguistics.

Oliver Hellwig, Salvatore Scarlata, Elia Ackermann, and Paul Widmer. 2020. The Treebank of Vedic Sanskrit. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5137–5146, Marseille, France. European Language Resources Association.

Amba Kulkarni, Pavankumar Satuluri, Sanjeev Panchal, Malay Maity, and Amruta Malvade. 2020. Dependency Relations for Sanskrit Parsing and Treebank. In *Proceedings of the 19th International Workshop on Treebanks and Linguistic Theories*, pages 135–150, Düsseldorf, Germany. Association for Computational Linguistics.

Chia-Wei Lin and Diego Luinetti. 2026. Building a Treebank for the Book of Ezra in the Old Georgian Oshki Bible. *Journal of Open Humanities Data*, 12(1):98.

Sebastian Nehrdich and Kurt Keutzer. 2026. MITRA: A Large-Scale Parallel Corpus and Multilingual Pretrained Language Model for Machine Translation and Semantic Retrieval for Pāli, Sanskrit, Buddhist Chinese, and Tibetan. *Preprint*, arXiv:2601.06400.

Rico Sennrich and Barry Haddow. 2016. Linguistic Input Features Improve Neural Machine Translation. In *Proceedings of the First Conference on Machine Translation: Volume 1, Research Papers*, pages 83–91, Berlin, Germany. Association for Computational Linguistics.

Milan Straka. 2018. UDPipe 2.0 Prototype at CoNLL 2018 UD Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207, Brussels, Belgium. Association for Computational Linguistics.

Klangjai Tammanam, Nuttachot Promrit, and Sajjaporn Waijanya. 2021. A hybrid approach to Pali Sandhi segmentation using BiLSTM and rule-based analysis. *Engineering and Applied Science Research*, 48(5):614–626.

Yuzuki Tsukagoshi and Ikki Ohmukai. 2025. Improving Classical Language Machine Translation using Supervised Fine-Tuning with Philological Commentary. In *Proceedings of the 39th Pacific Asia Conference on Language, Information and Computation*, pages 699–709, Hanoi, Vietnam. Association for Computational Linguistics.

Oskar von Hinüber. 1996. *A Handbook of Pāli Literature*. Walter de Gruyter.

Alexander Wynne. 2006. The Historical Authenticity of Early Buddhist Literature: A Critical Evaluation. *The Vienna Journal of South Asian Studies*, pages 35–70.

Dan Zigmond. 2023. Distinguishing Commentary from Canon: Experiments in Pāli Computational Linguistics. In *Proceedings of the Computational Sanskrit & Digital Humanities: Selected Papers Presented at the 18th World Sanskrit Conference*, pages 213–222. Association for Computational Linguistics.