Padamitra: Grounded Glossary Generation for Classical Sanskrit

Manoj Balaji Jagadeeshan **Sai Pragnaan Marala** **Pawan Goyal**

Indian Institute of Technology Kharagpur

Abstract

We introduce grounded glossary generation, a structured task requiring models to recover semantically meaningful Sanskrit phrases and produce translation-grounded meanings from a śloka - translation pair, formalizing the traditional pāṭha commentary practice as an evaluable NLP objective. We construct a benchmark of 31,316 śloka - translation glossary triples from the Vālmīki Rāmāyaṇa and Śrīmad Bhāgavatam, paired with two metrics: Jaccard for phrase recovery and Meaning Faithfulness for semantic consistency. Across zero-shot, few-shot, and instruction fine-tuned variants of Gemma-3n-E4B, Gemma-3-12B, Phi-4, and Qwen3.5-9B, instruction fine-tuning substantially outperforms prompting, while explicit segmentation yields gains. Error analysis identifies over-segmentation of sandhi and samāsa compounds as the dominant failure mode, pointing to morphological modeling as the key bottleneck for faithful Sanskrit lexical decomposition. Data[1] and Code[2] are available.

1 Introduction

Classical Sanskrit literature poses significant challenges for interpretation due to its morphological richness and flexible syntax. In particular, phonological fusion across word boundaries (*sandhi*) and systematic nominal compounding (*samāsa*) often produce surface forms whose segmentation into semantically meaningful units is itself a non-trivial task (Huet et al., 2009; Hellwig and Nehrdich, 2018). Classical commentators addressed this through *pāṭha* commentary - a glossary-like annotation that maps each resolved phrase in a verse (*śloka*) to its contextual meaning - a tradition that underlies modern pedagogical resources such as the IITK Valmiki Rāmāyaṇa[3] and Vedabase[4]. We formalize this tradition as **grounded glossary generation**: given a Sanskrit *śloka* s and its authoritative English translation t, the task is to produce a structured glossary $\mathcal{G} = \{(k_i, v_i)\}_{i=1}^{m}$, where each key k_i is a semantically coherent Sanskrit phrase and each value v_i is a meaning grounded in t. Crucially, grounding in the translation distinguishes this task from unconstrained lexicographic gloss generation: the meanings must be *faithful* to the provided translation, not merely plausible in isolation. This formulation makes interpretable lexical decomposition an explicit, evaluable objective rather than an opaque by-product of end-to-end translation.

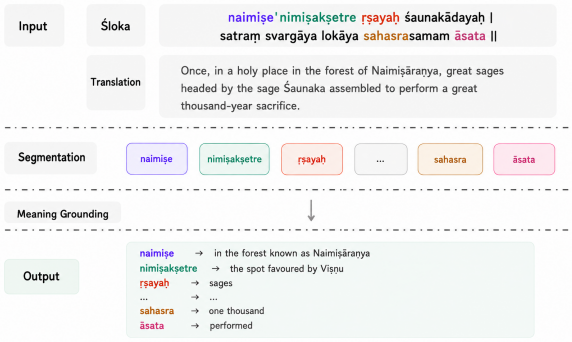


Figure 1: Grounded glossary generation. From a śloka-translation pair, the model recovers sandhi- and samāsa-resolved Sanskrit phrases and assigns each a meaning grounded in the provided translation, yielding a structured glossary.

Prior work in Sanskrit NLP has largely targeted individual sub-problems: sandhi splitting, morphological analysis, and segmentation (Huet et al., 2009; Kulkarni and Shukl, 2009; Krishna et al., 2017; Hellwig and Nehrdich, 2018); cross-lingual retrieval (Jagadeeshan et al., 2025); entity resolution (Sarkar et al., 2025); *anvaya* (Kr-

[1] https://huggingface.co/collections/sanganaka/padamitra-glossary-generation

[2] https://github.com/sanganaka-iitkgp/Padamitra-Glossary-Generation

[3] https://valmiki.iitk.ac.in

[4] https://vedabase.io/en/library/sb/

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 42745–42755
October 24-29, 2026 ©2026 Association for Computational Linguistics

ishna et al., 2019; Das et al., 2025); and poetry generation (Jagadeeshan et al., 2026a,b); Krishna et al. (2020) address several of these jointly in a single graph-based framework. Related work has also explored Sanskrit dependency parsing (Kulkarni and Shukl, 2009), Sanskrit–English neural machine translation (Aralikatte et al., 2021; Nehrdich et al., 2026), multilingual LLMs for low-resource languages (Ahuja et al., 2023), and bilingual word alignment using FastAlign (Dyer et al., 2013). However, no prior work directly addresses grounded phrase-level glossary generation as a unified task combining segmentation, semantic grounding, and lexicographic faithfulness.

Four gaps motivate this work. First, token-level aligners such as FastAlign cannot model Sanskrit's fusional morphology, where one surface token may span several glossary keys. Second, instruction-tuned LLMs handle structured prediction well via in-context learning (Brown et al., 2020; Wei et al., 2022), but their behavior on a task coupling sandhi resolution, boundary detection, and translation-grounded meaning assignment is untested. Third, MT metrics such as BLEU (Papineni et al., 2002) and chrF (Popović, 2015) measure sentence-level fluency, not phrase-level key recovery or per-entry semantic faithfulness. Fourth, whether explicit segmentation supervision helps over end-to-end learning is unestablished.

We address these gaps with a benchmark of 31,316 *śloka*–translation–glossary triples from the Vālmīki Rāmāyaṇa and Śrīmad Bhāgavatam, and two metrics: **Jaccard** for glossary key recovery and **Meaning Faithfulness** for translation-grounded semantic consistency. Across zero-shot, few-shot, and instruction fine-tuned variants of `gemma-3n-E4B-it`, `gemma-3-12b-it`, `phi-4` (14B), and `Qwen3.5-9B`, instruction fine-tuning (IFT) substantially outperforms prompting. Ablations show that predicted segmentation adds little, though gold segmentation raises Meaning Faithfulness from 0.787 to 0.872, and that IFT models degrade under word-order perturbation. Error analysis identifies over-segmentation of *sandhi* and *samāsa* compounds as the dominant failure mode.

2 Problem Statement

Given a Sanskrit śloka(s) and its English translation (t), the goal is to generate a grounded glossary $G = (k_i, v_i)_{i=1}^{m}$ where each k_i is a semantically meaningful Sanskrit phrase and v_i is its meaning grounded in the translation (t).

Unlike machine translation (Koehn et al., 2003), which targets fluent output over automatically induced phrase pairs, bilingual lexicon induction (Lample et al., 2018; Artetxe et al., 2017), which derives word-level equivalents from corpus-level statistics, or definition modeling (Noraset et al., 2017), which generates context-free glosses, grounded glossary generation jointly requires recovering morphologically complex phrase boundaries and producing meanings faithful to a specific provided translation. The generated glossary must remain grounded in the provided translation, capturing interpretable lexical units as structured phrase-meaning pairs while avoiding unsupported or hallucinated interpretations.

Example Illustration

Sanskrit śloka: नैमिषेऽनिमिषक्षेत्रे ऋषयः शौनकादयः ।
सत्रं स्वर्गायलोकाय सहस्रसममासत ॥
English Translation:
Once, in a holy place in the forest of Naimiṣāraṇya, great sages headed by the sage Śaunaka assembled to perform a great thousand-year sacrifice for the satisfaction of the Lord and His devotees.
Glossary:
नैमिषे → in the forest known as Naimiṣāraṇya अनिमिष-क्षेत्रे → the spot which is especially a favorite of Viṣṇu (who does not close His eyelids) ऋषयः → sages शौनक-आदयः → headed by the sage Śaunaka सत्रम् → sacrifice स्वर्गाय → the Lord who is glorified in heaven लोकाय → and for the devotees who are always in touch with the Lord सहस्र → one thousand, समम् → years आसत → performed.

3 Dataset

3.1 Data Sources

The dataset is curated from two publicly available educational resources: Rāmāyaṇa and Śrīmad Bhāgavatam, and use it for research purposes. Each instance contains:

(Sanskrit śloka, English translation, a glossary mapping Sanskrit phrases)

3.2 Data Preparation

We parse semi-structured glossary annotations into structured key-value dictionaries and apply light normalization, including Unicode and whitespace normalization, while preserving the original linguistic structure of the text.

The dataset is divided into train, validation, and

test splits: **Train:** 25,050, **Validation:** 3,133, **Test:** 3,133.

To reduce leakage, contiguous verses and near-duplicate glossary entries are prevented from appearing across splits. Overlaping gloss key between Train, Validation and Test is shown in Table 6.

4 Evaluation Metrics

4.1 Jaccard

To measure glossary key recovery, we compute an overlap-based Jaccard score between the set of generated glossary keys (G_p) and reference glossary keys (G_r):

$$\text{Jaccard} = \frac{|G_p \cap G_r|}{|G_p \cup G_r|}$$

where each glossary key corresponds to a Sanskrit phrase segment. This metric evaluates how accurately the model recovers the reference semantic units.

4.2 Meaning Faithfulness

Meaning Faithfulness measures semantic consistency between generated and reference glossary entries while penalizing segmentation errors in the keys.

For each generated key k_p, we find the closest reference key k_r by normalized Levenshtein similarity, computed with RapidFuzz[5] (Bachmann, 2025). We write this similarity as $\text{KeySim}(k_p, k_r) \in [0, 1]$ and treat the pair as matchable only above a threshold of 0.7; below it, $\text{KeySim} = 0$. For matched pairs, we compare the corresponding meanings using cosine similarity between BAAI/bge-base-en-v1.5 (Xiao et al., 2023) embeddings, obtained via SentenceTransformers (Reimers and Gurevych, 2019, 2020):

$$\text{ValSim}(v_p, v_r) = \cos\big(\phi(v_p), \phi(v_r)\big),$$

where $\phi(\cdot)$ is the embedding function. Each generated entry contributes the product

$$\text{Contribution} = \text{KeySim}(k_p, k_r) \cdot \text{ValSim}(v_p, v_r),$$

and the score for a śloka is the mean contribution over all generated entries; the reported score is the mean over the evaluation set. Averaging over *generated* entries penalizes spurious keys directly: an unmatched key contributes zero and drags the mean down. Section A.4 provides intuition behind the design of this metric.

[5] https://github.com/rapidfuzz/RapidFuzz

5 Methodology

5.1 Models Used

For the task of Glossary generation, we consider the models: `gemma-3n-E4B-it` (Team, 2025b), `gemma-3-12b-it` (Team, 2025a), `phi-4` (Abdin et al., 2024), `Qwen3.5-9B` (Qwen Team, 2026). The models translation capability is discussed in Section A.3.

5.2 Experimental Setup

Baselines. As a non-LLM baseline we use `FastAlign` (Dyer et al., 2013), a lightweight statistical word aligner based on IBM Model 2 (Brown et al., 1993) that learns token-level correspondences from parallel corpora; we extract the glossary from the alignments induced between each śloka and its English translation.

We additionally evaluate ByT5-Sanskrit (Nehrdich et al., 2024) in two configurations: (i) a single seq2seq model fine-tuned end-to-end to generate the glossary directly from the śloka and translation, and (ii) a two-stage pipeline in which one model is fine-tuned for segmentation (śloka → gloss keys) and a second generates the glossary from the predicted segments and the reference translation.

In-Context Learning: We experiment whether the a language model learn to generate the glossary with increased number of examples via zero-shot, few-shot learning with 1, 5 and 10 examples. These examples are available in Section A.9. The system and user prompt used for N-shot generation is available in Section A.7.

Instruction Fine-tuning (IFT): We instruction fine-tune language models using the aforementioned user prompt.

Settings for IFT is documented in Section A.2.

6 Result

6.1 Quantitative Analysis

Table 1 reports Jaccard and Meaning Faithfulness across all tuning and prompting regimes. IFT outperforms both `FastAlign` and in-context learning on every model and metric, with phi-4 best overall. Few-shot prompting improves over zero-shot but plateaus after one example, whereas instruction tuning yields larger and more stable gains — grounded glossary generation benefits from task-specific adaptation beyond token-level alignment or prompting. Meaning Faithfulness exceeds Jac-

Model	IFT	0-s	1-s	5-s	10-s
Jaccard					
FastAlign	.144		—		
ByT5 seq2seq	.543		—		
ByT5 pipeline	.520		—		
gemma-3n-E4B-it	.617	.175	.257	.285	.281
gemma-3-12b-it	.666	.238	.278	.304	.314
Qwen3.5-9B	.693	**.296**	**.377**	**.376**	**.378**
phi-4	**.716**	.092	.358	.230	.299
Meaning Faithfulness					
FastAlign	.320		—		
ByT5 seq2seq	.550		—		
ByT5 pipeline	.552		—		
gemma-3n-E4B-it	.746	.473	.557	.572	.565
gemma-3-12b-it	.768	.517	.557	.546	.566
Qwen3.5-9B	.781	**.570**	.631	**.628**	**.636**
phi-4	**.787**	.182	**.635**	.604	.628

Table 1: Glossary generation across tuning and prompting regimes. Jaccard measures key correctness; Meaning Faithfulness measures semantic accuracy of values against the ground truth. Both are averaged per entry. n-s = n-shot prompting. FastAlign and ByT5-Sanskrit are trained, not prompted, so few-shot settings do not apply. Leading zeros omitted; best per column in **bold**.

Error Category	Count
Over-Segmentation	122
General Segmentation Issues	22
Both Over and Under Segmentation	12
Under-Segmentation	9
Semantic & Translation Errors	4
No Issues	5

Table 2: Error analysis findings across 174 generated samples.

card throughout, indicating that models often produce semantically plausible meanings even when segmentation is imperfect. Section A.6 compares Jaccard against precision, recall, and F1.

6.2 Qualitative Analysis

We perform qualitative error analysis on 174 low-scoring samples (bottom 5% by Meaning Faithfulness) and categorize errors into six groups (Table 2). The dominant failure mode is *Over-Segmentation* (70%), where Sandhi and Samāsa compounds are incorrectly fragmented, e.g., `trasareṇuḥ`(त्रसरेणुः) split into `tra`(त्र) and `sareṇuḥ`(सरेणुः).

We also observe concurrent over- and under-segmentation, indicating inconsistent boundary detection, while purely semantic errors are comparatively rare. These results suggest that morphology-aware constraints and improved compound boundary modeling remain critical for Sanskrit glossary generation. A detailed analysis is available in Section A.5.

	Seen		Unseen	
Model	ZS	IFT	ZS	IFT
gemma-3n-E4B-it	30.65	80.94	17.93	54.98
gemma-3-12b-it	44.94	85.40	21.79	60.29
Qwen3.5-9B	**51.36**	86.30	**27.22**	64.86
phi-4	21.53	**87.85**	6.30	**66.45**
ByT5 seq2seq	—	43.33	—	28.24
ByT5 pipeline	—	68.68	—	49.29

Table 3: Gloss keys correctly predicted (%), split by whether the key was seen during instruction tuning (N = 12,194 seen, 8,956 unseen). ZS = zero-shot, IFT = instruction fine-tuned. ByT5-Sanskrit models are trained from scratch and have no zero-shot setting; *pipeline* segments before glossing. Best per column in **bold**.

6.3 Analysis on Seen vs Unseen Gloss Keys

Table 3 separates gloss keys seen during IFT (12,194) from those that were not (8,956). Gains are large on both: phi-4 improves by 66 points on seen keys and 60 points on unseen keys. Memorization of training units would predict little improvement on unseen keys; the near-parity instead indicates that the models learn generalizable segmentation and glossing behavior.

7 Ablations

We perform two ablation studies to analyze the role of explicit structural decomposition and robustness to Sanskrit word-order variation on the best performing model, `phi-4`.

7.1 Ablation 1: Segmentation-Augmented Glossary Generation

In this ablation, we instruction fine-tune models to first generate semantic Sanskrit segments and then produce grounded glossary meanings, in order to evaluate the impact of explicit intermediate segmentation. Table 4 results show that although performance is nearly identical to direct glossary generation for the models selected but it is seen that by providing the gold segmentations to generate only the glossary, we found that the Meaning Faithfulness score increased from 0.787 to 0.872. This indicates that segmentation quality plays an important role not only in recovering the correct key set, but also in improving the semantic faithfulness of the generated glossary.

Model	Jaccard	Meaning Faithfulness
Models that segment and gloss jointly		
gemma-3n-E4B-it	0.616	0.748
gemma-3-12b-it	0.656	0.769
Qwen3.5-9B	0.696	0.783
phi-4	**0.708**	**0.785**
phi-4 with external segmentation		
+ ByT5-Sanskrit (predicted)	0.501	0.741
+ gold segments *(oracle)*	1.000	*0.872*

Table 4: Performance of segmentation-augmented glossary generation. Models in the first block segment the śloka and generate meanings in a single pass; the second block feeds phi-4 externally produced segmentations. Best system result in **bold**; the gold-segment row is an oracle upper bound, not a comparable system.

7.2 Ablation 2: Robustness to Word Order Perturbation

Sanskrit preserves semantic relations through morphology, permitting relatively free word order. To test robustness to this variation, we randomly shuffle words within each śloka line, keeping the original line boundaries delimited by "।" and "॥".

Comparing Table 5 against Table 1, perturbation degrades every model, most severely `Qwen3.5-9B` (Jaccard $0.693 \rightarrow 0.394$). Even after adaptation, then, models rely on sequential cues rather than the morphological marking.

These results also speak to contamination. The pre-trained models may have seen publicly available Sanskrit texts and analyses, but weak zero-shot performance and the large gains from instruction fine-tuning indicate task-specific adaptation rather than retrieval of memorized material.

Model	Jaccard	Meaning Faithfulness
gemma-3-12b-it	0.587	0.750
phi-4	**0.632**	**0.768**
Qwen3.5-9B	0.394	0.682

Table 5: Performance under word-order perturbation: words within each śloka line are randomly shuffled.

8 Conclusion

We introduce grounded glossary generation as a structured task requiring recovery of semantically meaningful Sanskrit phrases and translation-grounded glossary meanings. We further present a benchmark dataset, evaluation framework, and systematic study of large language models, showing that instruction fine-tuning substantially improves glossary generation while explicit segmentation supervision affect and word-ordering does affect the performance.

Our results identify morphology and compound boundary detection as the primary bottlenecks for Sanskrit glossary generation. Future work can explore morphology-aware modeling, retrieval-grounded generation, and multilingual lexicographic systems for low-resource settings.

Limitations

- **Corpus coverage.** Our data is drawn from two texts, the Vālmīki Rāmāyaṇa and Śrīmad Bhāgavatam. Both are narrative *itihsa-pura* works with comparatively regular syntax; results may not transfer to traditions with distinct compositional styles, such as ornate Kāvya or Vedic prose.
- **Metric sensitivity.** Jaccard relies on surface string matching and Meaning Faithfulness on embedding similarity. Neither recognizes morphological equivalence between valid sandhi resolutions, so orthographically distinct but correct segmentations are penalized.
- **Training budget.** Instruction tuning ran under memory constraints with reduced batch sizes and shorter sequence lengths, and all models used LoRA rather than full fine-tuning. Reported IFT scores are therefore a lower bound on what the approach can achieve, and whether full fine-tuning closes the remaining segmentation gap is untested.
- **Error analysis scope.** We analyze only the bottom 5% of samples by Meaning Faithfulness. Failure modes in the mid-range of the distribution, where most errors occur, may differ.

Acknowledgments

We thank Kartikeya Singh (Indian Institute of Technology Kharagpur) for his contributions to preliminary work on this project. This work was supported in part by the GCP Research Grant for Gemma and by the National Language Translation Mission (Bhashini), Government of India. We thank Svarupa[6] for providing two NVIDIA L40 GPUs for this research. We also thank the anonymous reviewers for feedback that improved the paper.

[6] https://svarupa.org/

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *Preprint*, arXiv:2412.08905.

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Mohamed Ahmed, and 1 others. 2023. Mega: Multilingual evaluation of generative ai. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4232–4267.

Rahul Aralikatte, Miryam de Lhoneux, Anoop Kunchukuttan, and Anders Søgaard. 2021. Itihasa: A large-scale corpus for Sanskrit to English translation. In *Proceedings of the 8th Workshop on Asian Translation (WAT2021)*, pages 191–197, Online. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Max Bachmann. 2025. rapidfuzz/rapidfuzz: Release 3.13.0.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Kunal Kingkar Das, Manoj Balaji Jagadeeshan, Nallani Chakravartula Sahith, Jivnesh Sandhan, and Pawan Goyal. 2025. Still not there: Can LLMs outperform smaller task-specific Seq2Seq models on the poetry-to-prose conversion task? In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 343–357, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Oliver Hellwig and Sebastian Nehrdich. 2018. Sanskrit word segmentation using character-level recurrent and convolutional neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2754–2763, Brussels, Belgium. Association for Computational Linguistics.

Gérard Huet, Amba Kulkarni, and Peter Scharf. 2009. *Sanskrit computational linguistics*, volume 5402. Springer.

Manoj Balaji Jagadeeshan, Samarth Bhatia, Pretam Ray, Harshul Raj Surana, Akhil Rajeev P, Priya Mishra, Annarao Kulkarni, Ganesh Ramakrishnan, Prathosh Ap, and Pawan Goyal. 2026a. Chandomitra: Towards generating structured Sanskrit poetry from natural language inputs. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 518–534, Rabat, Morocco. Association for Computational Linguistics.

Manoj Balaji Jagadeeshan, Prince Raj, and Pawan Goyal. 2025. Anveshana: A new benchmark dataset for cross-lingual information retrieval on English queries and Sanskrit documents. In *Computational Sanskrit and Digital Humanities - World Sanskrit Conference 2025*, pages 161–180, Kathmandu, Nepal. Association for Computational Lingustics.

Manoj Balaji Jagadeeshan, Atul Singh, Nallani Chakravartula Sahith, Amrith Krishna, and Pawan Goyal. 2026b. Pingala: Prosody-aware decoding for sanskrit poetry generation. *Preprint*, arXiv:2603.24413.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 48–54.

Amrith Krishna, Bishal Santra, Ashim Gupta, Pavankumar Satuluri, and Pawan Goyal. 2020. A graph-based framework for structured prediction tasks in Sanskrit. *Computational Linguistics*, 46(4):785–845.

Amrith Krishna, Pavan Kumar Satuluri, and Pawan Goyal. 2017. A dataset for Sanskrit word segmentation. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 105–114, Vancouver, Canada. Association for Computational Linguistics.

Amrith Krishna, Vishnu Sharma, Bishal Santra, Aishik Chakraborty, Pavankumar Satuluri, and Pawan Goyal. 2019. Poetry to prose conversion in Sanskrit as a linearisation task: A case for low-resource languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1160–1166, Florence, Italy. Association for Computational Linguistics.

Amba Kulkarni and Devanand Shukl. 2009. Sanskrit morphological analyser: Some issues. *Indian Linguistics*, 70(1-4):169–177.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Sebastian Nehrdich, David Allport, Sven Sellmer, Jivnesh Sandhan, Manoj Balaji Jagadeeshan, Pawan Goyal, Sujeet Kumar, and Kurt Keutzer. 2026. Mitrasamgraha: A comprehensive classical sanskrit machine translation dataset. *Preprint*, arXiv:2601.07314.

Sebastian Nehrdich, Oliver Hellwig, and Kurt Keutzer. 2024. One model is all you need: ByT5-Sanskrit, a unified model for Sanskrit NLP tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13742–13751, Miami, Florida, USA. Association for Computational Linguistics.

Thanapon Noraset, Chen Liang, Larry Birnbaum, and Doug Downey. 2017. Definition modeling: learning to define word embeddings in natural language. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 32593266. AAAI Press.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Sujoy Sarkar, Gourav Sarkar, Manoj Balaji Jagadeeshan, Jivnesh Sandhan, Amrith Krishna, and Pawan Goyal. 2025. Mahānāma: A unique testbed for literary entity discovery and linking. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24970–24984, Suzhou, China. Association for Computational Linguistics.

Gemma Team. 2025a. Gemma 3.

Gemma Team. 2025b. Gemma 3n.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

A Appendix

A.1 Dataset Gloss Key Overlap between Train, Validation and Test

Split	Unique Gloss Keys	Seen in Train	Percentage
Train	101,573	-	-
Validation	21,048	12,073	**57.36%**
Test	21,150	12,194	**57.65%**

Table 6: Dataset Gloss Key Overlap Information

A.2 Hyperparameters used for LLM

All models were fine-tuned using LoRA with largely shared hyperparameters. Common settings included LoRA scaling ($\alpha = 16$), dropout (0.05), RS-LoRA, AdamW fused optimization, a learning rate of 3×10^{-5}, cosine scheduling with 0.15 warmup, BF16 precision, gradient clipping (1.0), and training for 10 epochs. Checkpointing and evaluation were performed periodically, with the best model selected using validation loss.

Model-specific differences mainly involved LoRA rank, target modules, batch size, and sequence length. `gemma-3-12b-it` used $r = 128$ with adaptation on all linear layers, while `phi-4` and `Qwen3.5-9B` used $r = 256$. `Qwen3.5-9B` employed assistant_only_loss, whereas the others used completion_only_loss. `gemma-3n-E4B-it` applied LoRA only to attention and feed-forward projection layers, used smaller batch sizes (2×4) for per-device batch size and gradient accumulation. Due to memory constraints, we disabled gradient checkpointing, and used shorter input sequences (512 tokens) compared to `gemma-3-12b-it` where we used (4×4) while `phi-4` and `Qwen3.5-9B` used (8×2).

A.3 Evaluating selected models Translation capability

According to Table 7, the translation-quality ranking tracks the Jaccard ranking: Phi-4 leads both; Qwen, Gemma-3, and Gemma-3n cluster in the middle. This confirms the hypothesis: *how well the model was exposed to Sanskrit data in its pre-training and to its ability to understand the Sanskrit-to-English translation task shedding further light on the relative strengths of these models*, rather than challenging our contribution i.e. Grounded Glossary Generation performance connects to pre-training Sanskrit exposure, a useful diagnostic property of the benchmark. It does not alter the our core claim that segmentation is the dominant bottleneck, which the ground-truth-substitution experiment isolates independently of translation ability.

Model	BLEU	ChrF	Semantic Similarity
Phi-4	6.21	30.61	73.77%
Qwen3.5-9B	4.96	28.11	70.20%
Gemma-3n-E4B-it	4.93	27.70	70.57%
Gemma-3-12B-it	4.69	27.31	70.83%

Table 7: Comparison of model performance across BLEU, ChrF, and semantic similarity.

A.4 Intuition behind the design of Meaning Faithfulness

The intuition behind this design of metric is to explicitly handle three common scenarios in key generation: under segmentation, over segmentation and proper segmentation.

Under Segmentation: In this case, the model fails to split a word that should have been fragmented. Since the ground-truth key is segmented, the maximum Levenshtein similarity between the generated key and the ground truth keys typically decreases. Multiplying this lower KeySim with the cosine similarity therefore penalizes the generated key.

Over segmentation: Here, the model splits a key into fragments more than necessary. As in the previous case, the resulting fragments obtain lower Levenshtein similarity scores with the ground-truth keys, and multiplying by KeySim reduces their contribution, thereby penalizing the model.

In both of the above cases, severe under segmentation or over segmentation may cause the similarity score to fall below the threshold, setting KeySim to 0. (For example, this would happen if the model splits a single word into three fragments or merges three separate words into one.)

Proper segmentation: When the key is segmented correctly, KeySim remains high, 100% for an exact match and close to 100% for near perfect matches. In such cases, the key is effectively not penalized.

A.5 Qualitative Analysis of Phi-4

To better understand model failures beyond quantitative metrics, we conducted a qualitative error analysis on 174 representative samples (bottom 5% by Meaning Faithfulness score). We grouped the errors into six mutually exclusive categories based on mismatches between the reference and generated glossary entries (Table 2).

The most frequent issue was *Over-Segmentation* (70% of cases), where compounds formed through Sandhi and Samasa were fragmented into smaller units, leading to loss of contextual meaning. For example, trasareṇuḥ(त्रसरेणुः) was incorrectly split into tra(त्र) and sareṇuḥ(सरेणुः).

We also observed cases of *Concurrent Over and Under Segmentation*, where the model simultaneously failed to split some compounds while over-splitting adjacent phrases, indicating inconsistent boundary detection in longer Sanskrit strings. *Under-Segmentation* was less common but still harmful, as in puruṣa-paricaryay=a(पुरुष-परिचर्यया), which remained unsplit despite the reference expecting finer segmentation.

A smaller subset of errors were purely *Semantic*, where the generated gloss was contextually incorrect, such as translating ambaṣṭha(अम्बष्ठ) as 'driver" instead of 'elephant-keeper."

For the 5 no issues samples, there were no issues on the generated output but rather the key:val pairs were wrongly translated in the ground truth glossary.

Overall, these findings highlight the need for stronger morphological constraints and improved boundary detection to reduce over-fragmentation in Sanskrit glossary generation.

A.6 Comparing Jaccard against Precision, Recall, F1-score

Precision, Recall, are also valid alternatives for evaluating glossary-unit recovery. We chose Jaccard because the task is formulated as recovering the set of glossary units, and Jaccard provides a simple set-overlap measure that jointly penalizes missing and spurious units. Since Jaccard and F1 are closely related monotonic measures, we do not expect the relative ranking of models to change substantially which is also observed in Table 8. Similarly for Ablation 1, the Jaccard vs Precision, Recall and F1-score, the respective scores are available in Table 9.

Model	Precision	Recall	F1-score	Jaccard
Phi-4	0.8145	0.8237	0.8175	0.716
Qwen3.5-9B	0.8016	0.8041	0.8012	0.693
Gemma-3-12B-it	0.7775	0.7875	0.7807	0.666
Gemma-3n-E4B-it	0.7439	0.7453	0.7426	0.617

Table 8: The metric values for our test data, for correct gloss key identification

Model	Precision	Recall	F1-score	Jaccard
Phi-4	0.8102	0.8173	0.8120	0.708
Qwen3.5-9B	0.8055	0.8042	0.8031	0.696
Gemma-3-12B-it	0.7770	0.7732	0.7732	0.656
Gemma-3n-E4B-it	0.745	0.742	0.743	0.616

Table 9: Ablation 1 metric values for our test data, for correct gloss key identification

A.7 Promtps used for N-shot and IFT

System Prompt

You are a Sanskrit-to-English glossary extraction engine. Given a Sanskrit shloka and its English translation, output a single JSON object mapping each Sanskrit word to its English meaning.
Rules:
1. Output ONLY a Python dict. Start with "{", end with "}". No other text.
2. Keys are Sanskrit words (or sandhi-resolved tokens) as they appear in the shloka.
3. Values are their English meanings as evidenced by the translation.
4. No markdown, no code fences, no explanation, no preamble.

Format: {"glossary": {"sanskrit_word": "english_meaning", ...}}

User Prompt

Generate a Sanskrit-English glossary mapping for the following shloka based on its translation.
Shloka: {shloka}
Translation: {translation}

A.8 Prompts used for Ablation 1

System Prompt for Ablation 1

Perform Padaccheda (resolve sandhi, keep samasa compound words intact). Analyze the shloka word by word using the translation as a reference. End your reasoning with the resolved word sequence as: [word1, word2, ...].

STEP 2 FINAL GENERATION:
Output the glossary as a code block. Rules for the output:
1. Output exactly ONE valid Python dict with a single key 'glossary'.
2. All keys are the Padaccheda-resolved Sanskrit words, in the order they appear in the shloka.
3. All values are their English meanings, derived strictly from the provided translation.
4. No trailing commas.
5. No extra keys, no nested objects flat dictionary only.

RESPONSE FORMAT (follow exactly, do not deviate):
REASONING:
<your padaccheda analysis ending with the resolved word list>

FINAL GENERATION:
{"glossary": {"sanskrit_word": "english_meaning", ...}}
Do NOT include any text outside these two sections. Do NOT add greetings, notes, or explanations after the output block.

User Prompt for Ablation 1 IFT

INPUT: You are an expert in Sanskrit. Perform Padaccheda (word resolution) on the following Sanskrit shloka, then generate the glossary mapping using the generated Padaccheda and provided English translation.
Shloka: <shloka>
Translation: <translation>

A.9 N-shot Examples

The following data was used as examples for all N-shot experiments. For 1-shot, the first example, for 5-shot, the first 5 examples and similarly for 10-shot all the 10 examples were used.

Shloka: न वयं क्लेशबीजानि यत: स्यु: पुरुषर्षभ । पुरुषं तं विजानीमो वाक्यभेदविमोहिता: ।
Translation: O greatest among human beings, it is very difficult to ascertain the particular miscreant who has caused our sufferings, because we are bewildered by all the different opinions of theoretical philosophers.
Glossary: {'न': 'not', 'वयम्': 'we', 'क्लेश-बीजानि': 'the root cause of sufferings', 'यतः': 'wherefrom', 'स्युः': 'it so happens', 'पुरुष-ऋषभ': 'O greatest of all human beings', 'पुरुषम्': 'the person', 'तम्': 'that', 'विजानीमः': 'know', 'वाक्य-भेद': 'difference of opinion', 'विमोहिताः': 'bewildered by'}

Shloka: सोऽधिक्षिप्तो दुर्वाचोभि: पदाहत इवोरग: । निश्चक्राम गदापाणिरमर्षात्ताम्रलोचन: ।
Translation: Offended by these harsh words, śambara became as agitated as a kicked snake. He came out, club in hand, his eyes red with rage.
Glossary: {'सः': 'he', 'अधिक्षिप्तः': 'insulted', 'दुर्वाचोभिः': 'by harsh words', 'पदा': 'by a foot', 'आहतः': 'struck', 'इव': 'like', 'उरगः': 'a snake', 'निश्चक्राम': 'came out', 'गदा': 'a club', 'पाणिः': 'in his hand', 'अमर्षात्': 'out of intolerant anger', 'ताम्र': 'copper-red', 'लोचनः': 'whose eyes'}

Shloka: दृष्ट्वान्यांश्च महोत्पातानतत्तत्त्वविद: प्रजा: । ब्रह्मपुत्रानृते भीता मेनिरे विश्वसम्प्लवम् ।
Translation: Marking these and many other omens of evil times, everyone but the four sage sons of Brahmā, who were aware of the fall of Jaya and Vijaya and of their birth as Diti's sons, was seized with fear. They did not know the secrets of these portents and thought that the dissolution of the universe was at hand.
Glossary: {'दृष्ट्वा': 'having seen', 'अन्यान्': 'others', 'च': 'and', 'महा': 'great', 'उत्पातान्': 'evil omens', 'अ-तत्-तत्त्व-विदः': 'not knowing the secret (of the portents)', 'प्रजाः': 'people', 'ब्रह्म-पुत्रान्': 'the sons of Brahmā (the four Kumāras)', 'ऋते': 'except', 'भीताः': 'being fearful', 'मेनिरे': 'thought', 'विश्व-सम्प्लवम्': 'the dissolution of the universe'}

Shloka: कथं वर्तेत विहरेत् कैर्वा ज्ञायेत लक्षणै: । किं भुञ्जीतोत विसृजेच्छयीतासीत याति वा । एतदच्युत मे ब्रूहि प्रश्नं प्रश्नविदां वर । नित्यबद्धो नित्यमुक्त एक एवेति मे भ्रम: ।
Translation: O my Lord, Acyuta, the same living entity is sometimes described as eternally conditioned and at other times as eternally liberated. I am not able to understand, therefore, the actual situation of the living entity. You, my Lord, are the best of those who are expert in answering philosophical questions. Please explain to me the symptoms by which one can tell the difference between a living entity who is eternally liberated and one who is eternally conditioned. In what various ways would they remain situated, enjoy life, eat, evacuate, lie down, sit or move about?
Glossary: {'कथम्': 'in what way', 'वर्तेत': 'he is situated', 'विहरेत्': 'he enjoys', 'कैः': 'by which', 'वा': 'or', 'ज्ञायेत': 'would be known', 'लक्षणैः': 'by symptoms', 'किम्': 'what', 'भुञ्जीत': 'he would eat', 'उत': 'and', 'विसृजेत्': 'would evacuate', 'शयीत': 'would lie down', 'आसीत': 'would sit', 'याति': 'goes', 'एतत्': 'this', 'अच्युत': 'O Acyuta', 'मे': 'to me','ब्रूहि': 'explain', 'प्रश्नम्': 'the question', 'प्रश्न-विदाम्': 'of all those who know how to answer questions', 'वर': 'O the best', 'नित्य-बद्धः': 'eternally conditioned', 'नित्य-मुक्तः': 'eternally liberated', 'एकः': 'singular', 'एव': 'certainly', 'इति': 'thus', 'भ्रमः': 'confusion'}

Shloka: तत्रेतिकृत्यमुपशिक्ष यथोपदेशं येनैष मे कर्शितोऽतिरिरंसयात्मा । सिद्ध्येत ते कृतमनोभवधर्षिताया दीनस्तदीश भवनं सदृशं विचक्ष्व ।
Translation: Devahūti continued: My dear lord, I am

struck by excited emotion for you. Therefore kindly make what arrangements must be made according to the scriptures so that my skinny body, emaciated through unsatisfied passion, may be rendered fit for you. Also, my lord, please think of a suitable house for this purpose.
Glossary: {'तत्र': 'in that', 'इति-कृत्यम्': 'what is necessary to be done', 'उपशिक्ष': 'perform', 'यथा': 'according to', 'उपदेशम्': 'instruction in scripture', 'येन': 'by which', 'एषः': 'this', 'मे': 'my', 'कर्शितः': 'emaciated', 'अतिरिरꣳ-सया': 'due to intense passion not being satisfied', 'आत्मा': 'body', 'सिद्ध्येत': 'it may be rendered fit', 'ते': 'for you', 'कृत': 'excited', 'मनः-भव': 'by emotion', 'धर्षितायाः': 'who am struck', 'दीनः': 'poor', 'तत्': 'therefore', 'ईश': 'O my dear lord', 'भवनम्': 'house', 'सदृशम्': 'suitable', 'विचक्ष्व': 'please think of'}

Shloka: न वेद धर्मं किल पद्मयोनि- र्न ब्रह्मपुत्रा भृगुनारदाद्याः । न वै कुमारः कपिलो मनुश्च ये नो निषेधन्त्यतिवर्तिनं हरम् ।
Translation: Alas, Lord Brahmā, who has taken his birth from the lotus flower, does not know the principles of religion, nor do the great saints like Bhgu and Nārada, nor the four Kumāras, headed by Sanat-kumāra. Manu and Kapila have also forgotten the religious principles. I suppose it to be because of this that they have not tried to stop Lord śiva from behaving improperly.
Glossary: { 'न': 'not', 'वेद': 'knows', 'धर्मम्': 'the religious principles', 'किल': 'indeed', 'पद्म-योनिः': 'Lord Brahmā', 'ब्रह्म-पुत्राः': 'the sons of Lord Brahmā', 'भृगु': 'Bhgu', 'नारद': 'Nārada', 'आद्याः': 'and so on', 'वै': 'indeed', 'कुमारः': 'the four Kumāras (Sanaka', 'कपिलः': 'Lord Kapila', 'मनुः': 'Manu himself', 'च': 'and', 'ये': 'who', 'नो': 'not', 'निषेधन्ति': 'order to stop', 'अति-वर्तिनम्': 'who is beyond laws and orders', 'हरम्': 'Lord śiva'}

Shloka: कर्माण्यारभते येन पुमानिह विहाय तम् । अमुत्रान्येन देहेन जुष्टानि स यदश्नुते ।
Translation: The results of whatever a living entity does in this life are enjoyed in the next life.
Glossary: {'कर्माणि': 'fruitive activities', 'आरभते': 'begins to perform', 'येन': 'by which', 'पुमान्': 'a living entity', 'इह': 'in this life', 'विहाय': 'giving up', 'तम्': 'that', 'अमुत्र': 'in the next life', 'अन्येन': 'another', 'देहेन': 'by a body', 'जुष्टानि': 'the results', 'सः': 'he', 'यत्': 'that', 'अश्नुते': 'enjoys'}

Shloka: आरोप्यारुरुहे यानं भ्रातृभ्यां हनुमद्युतः । विभीषणाय भगवान् दत्त्वा रक्षोगणेशताम् । लङ्कामायुश्च कल्पान्तं ययौ चीर्णव्रतः पुरीम् ।
Translation: After giving Vibhīaa the power to rule the Rākasa population of Laṅkā for the duration of one kalpa, Lord Rāmacandra, the Supreme Personality of Godhead [Bhagavān], placed Sītādevī on an airplane decorated with flowers and then got on the plane Himself. The period for His living in the forest having ended, the Lord returned to Ayodhyā, accompanied by Hanumān, Sugrīva and His brother Lakmaa.
Glossary: {'आरोप्य': 'keeping or placing', 'आरुरुहे': 'got up', 'यानम्': 'on the airplane', 'भ्रातृभ्याम्': 'with His brother Lakmaa and the commander Sugrīva', 'हनुमत्-युतः': 'accompanied by Hanumān', 'विभीषणाय': 'unto Vibhīaa', 'भगवान्': 'the Lord', 'दत्त्वा': 'gave charge', 'रक्षः-गण-ईशताम्': 'the power to rule over the Rākasa population of Laṅkā', 'लङ्काम्': 'the state of Laṅkā', 'आयुः च': 'and the duration of life', 'कल्प-अन्तम्': 'for many', 'ययौ': 'returned home', 'चीर्ण-व्रतः': 'finishing the duration of time living in the forest', 'पुरीम्': 'to Ayodhyā-purī'}

Shloka: एतां विद्यामधिगतो विश्वरूपाच्छतक्रतुः । त्रैलोक्यलक्ष्मीं बुभुजे विनिर्जित्य मृधेऽसुरान् ।
Translation: King Indra, who performed one hundred sacrifices, received this prayer of protection from Viśvarūpa. After conquering the demons, he enjoyed all the opulences of the three worlds.
Glossary: {'एताम्': 'this', 'विद्याम्': 'prayer', 'अधिगतः': 'received', 'विश्वरूपात्': 'from the brāhmaa Viśvarūpa', 'शत-क्रतुः': 'Indra', 'त्रैलोक्य-लक्ष्मीम्': 'all the opulence of the three worlds', 'बुभुजे': 'enjoyed', 'विनिर्जित्य': 'conquering', 'मृधे': 'in battle', 'असुरान्': 'all the demons'}

Shloka: ननृतुस्तस्य पुरतः स्त्रियोऽथो गायका जगुः । मृदङ्गवीणापणवैर्वाद्यं चक्रुर्मनोरमम् ।
Translation: The women danced before the sage, and the celestial singers sang to the charming accompaniment of drums, cymbals and vīās.
Glossary: {'ननृतुः': 'danced', 'तस्य': 'of him', 'पुरतः': 'in front', 'स्त्रियः': 'women', 'अथ उ': 'and furthermore', 'गायकाः': 'singers', 'जगुः': 'sang', 'मृदङ्ग': 'with drums', 'वीणा': 'stringed instruments', 'पणवैः': 'and cymbals', 'वाद्यम्': 'instrumental music', 'चक्रुः': 'they made', 'मनः-रमम्': 'charming'}