Automatic Accent Restoration in Vedic Sanskrit with Neural Language Models

Yuzuki Tsukagoshi and Ikki Ohmukai
The University of Tokyo / Tokyo, Japan
yuzuki@l.u-tokyo.ac.jp

Abstract

Vedic Sanskrit, the oldest attested form of Sanskrit, employs a distinctive pitch-accent system that marks one syllable per word. To the best of our knowledge, this work presents the first application of large language models to the automatic restoration of accent marks in transliterated Vedic Sanskrit texts. A comprehensive corpus was assembled by extracting major Vedic works from the TITUS project and constructing paired samples of unaccented input and correctly accented references, yielding more than 100,000 training examples. Three generative LLMs were fine-tuned on this corpus: a LoRA-adapted Llama 3.1 8B Instruct model, OpenAI GPT-4.1 nano, and Google Gemini 2.5 Flash. These models were trained in a sequence-to-sequence fashion to insert accent marks at appropriate positions. Evaluation on roughly 2,000 sentences using precision, recall, F1, character error rate, word error rate, and ChrF1 metrics shows that fine-tuned models substantially outperform their untuned baselines. The LoRA-tuned Llama achieves the highest F1, followed by Gemini 2.5 Flash and GPT-4.1 nano. Error analysis reveals that the models learn to infer accents from grammatical and phonological context. These results demonstrate that LLMs can capture complex accentual patterns and recover lost information, opening possibilities for potential improvements in sandhi splitting, morphological analysis, syntactic parsing and machine translation in Vedic NLP pipelines.

1 Introduction

Vedic Sanskrit is the oldest attested form of Sanskrit and preserves the religious and philosophical contexts of ancient India. Vedic Sanskrit texts are distinguished by a pitch accent system that marks one syllable per word as accented. The accent marks are essential for linguistic and philological analysis of the Vedas. Accurate accentuation can signal morphological and syntactic information in Vedic Sanskrit, which differs significantly from Classical Sanskrit. However, some Vedic texts lack accent notations, and restoring Vedic accent marks has received little attention in natural language processing to date. This is a challenging sequence prediction task: the accent of a word is not always predictable from its surface form alone; it often depends on the grammatical context.

In this work, we address the task of automatic Vedic accent restoration using modern large language models (LLMs). We fine-tune three state-of-the-art models on a comprehensive Vedic corpus: (1) a LoRA-adapted Llama 3.1 8B Instruct model, (2) an OpenAI GPT-4.1 nano model, and (3) a Google Gemini 2.5 Flash model via supervised fine-tuning (SFT). We avoid older sequence-to-sequence-based or BERT-like models, focusing instead on these generative LLMs which can directly produce accented text. Our contributions include:

- assembling a large accented Vedic corpus from the TITUS project and constructing pairs of accented and unaccented sentences; [1]
- demonstrating efficient fine-tuning of open and closed large language models on this task; and
- evaluating the models' performance on accent restoration using standard precision, recall, F1 metrics, CER, WER, and ChrF1.

We show that all models achieve high accuracy in restoring Vedic accent marks.

Our results represent the first application of large-scale language models to the Vedic accent restoration problem. By accurately reconstructing

[1] TITUS (Thesaurus Indogermanischer Text- und Sprachmaterialien) provides digitized Indo-European texts. https://titus.uni-frankfurt.de/indexe.htm?/texte/texte2.htm.

Proceedings of the 1st Workshop on Benchmarks, Harmonization, Annotation, and Standardization for Human-Centric AI in Indian Languages (BHASHA 2025), pages 83–90

accentual patterns, the models effectively bridge a gap in Sanskrit digitization efforts. This capability indicates that Vedic accent restoration task could potentially support downstream tasks, such as sandhi splitting, morphological analysis, syntactic parsing, and machine translation, although systematic empirical vertification is left for future work.

2 Vedic Accent System

Vedic Sanskrit is the oldest attested variety of Sanskrit, and its distinctive accent system sets it apart from later stages of the language. In Latin transliteration, accent marks are represented by the acute and the grave accents.

The fundamental rule of Vedic accentuation is that each word carries only one accent. There are, however, several exceptions, including enclitics, finite verbs in main clauses, vocatives and other conditions (Macdonell, 1910).

Nouns, adjectives, and verbs in Vedic Sanskrit inflect according to their semantic roles. Some paradigms exhibit a dynamic accent system in which the accent position changes across inflected forms. For example, the active present participle of the verb *as* 'to be' shows a nominative singular form *s-án*, with the accent on the suffix, and a genitive singular form *s-at-ás*, with the accent on the ending.

The position of the accent is also crucial in determining the meaning of compounds. Vedic Sanskrit has a rich system of compound formation, including: two endocentric types, determinative (*Tatpuruṣa*) and descriptive (*Karmadhāraya*); an exocentric, possessive type (*Bahuvrīhi*); a copulative type (*Dvandva*); an iterative type (*Āmredita*); prepositional governing compounds; syntactic compounds; and complexive compounds (Gotō, 2013). The position of the accent helps to distinguish compound types. *Tatpuruṣa* and *Karmadhāraya*, which are endocentric compounds, typically bear the accent on the final member, whereas *Bahuvrīhi*, which is exocentric, has the accent on the first member.

3 Related Works

This research explores the task of restoring Vedic Sanskrit accent. Though accents are critical in Vedic Sanskrit, the present work is the first to frame their recovery as an NLP task. Related studies fall into three broad areas: computational analyses of Vedic accentuation, diacritic or accent restoration in modern languages, and automatic restoration of damaged ancient texts.

Computational modeling of Vedic accent: Scholars have long noted that Vedic Sanskrit accent cannot be predicted by simple syllable count or phonological weight. Sandell (2024) argues that stress assignment relies on morphological structure and prosody rather than arbitrary lists of accented affixes. Instead of positing separate phonological strata or "dominant" affixes, Sandell proposes a uniform Optimality Theory analysis where each morpheme enters the derivation with its own foot structure; accent emerges from the interaction of faithfulness to morphological heads and markedness constraints. This approach achieves computational uniformity across stems and suffixes and avoids listing stem-specific accent patterns. Such theoretical work provides insights into how morphological context might inform machine learning models for accent restoration.

Accent and diacritic restoration in modern languages:

Romance languages

Yarowsky (1994) treats diacritic restoration in Spanish and French as a lexical ambiguity resolution problem. Omission of diacritics (e.g. acute or grave accents) produces many homographs, causing lexical and syntactic ambiguity. Each unaccented surface form has a set of possible accented lemmas, and the task is to choose the correct one using context. The proposed statistical decision-list algorithm selects the single most informative contextual feature, rather than combining multiple cues, to choose the correct accent. This simple method achieves over 99% accuracy on both languages, demonstrating that moderate training data and local context can resolve diacritic ambiguity with high precision.

Arabic

Aldallal et al. (2025) build a compact decoder-only Transformer model (SADEED) with about 140M parameters for Arabic diacritization. Modern Arabic is typically written without short vowel marks (ḥarakāt), making diacritization necessary for unambiguous parsing, text-to-speech and machine translation. The task is challenging because Arabic exhibits rich morphology, multiple registers (Classical vs. Modern Standard Arabic), and limited diacritized corpora. Trained on a new benchmark corpus (SadeedDiac-25) combining modern and classical texts, their model deliv-

ers competitive accuracy while being much smaller than prior systems. Their work highlights the importance of specialized datasets and demonstrates that carefully designed, lightweight models can yield strong diacritization performance.

Vietnamese

Dang and Nguyen (2020) propose a hybrid model combining a Transformer decoder with a diacritic penalty layer for Vietnamese diacritic restoration. Vietnamese uses tone marks and other diacritics on most words, nearly 90% of words contain diacritics, and over 80% of these have multiple possible tonal reconstructions. Restoration is therefore indispensable for downstream applications but challenging because sequence-to-sequence neural models can generate invalid syllables and are slow. In their method, the decoder outputs one character at a time, while the penalty layer restricts outputs to valid diacritic letters. This reduces processing time by roughly eight to ten times compared with beam search and preserves or slightly improves F1-score relative to state-of-the-art sequence-to-sequence models. Their approach shows that explicit constraints on output vocabulary can improve both efficiency and accuracy in diacritic restoration.

Ancient text restoration:

Assael et al. (2019) introduce PYTHIA, the first deep-learning system for restoring damaged ancient Greek inscriptions. Ancient inscriptions often survive only fragmentarily, requiring specialists to hypothesize missing text. After constructing the PHI-ML corpus from the Packard Humanities Institute's Greek epigraphic collection, the authors train a model that jointly leverages character-level and word-level information to predict missing characters. On this dataset, PYTHIA's predictions reduce the character error rate to 30.1%, compared with 57.3% for human epigraphists, and the correct sequence appears within the top-20 hypotheses in 73.5% of cases.

4 Dataset

4.1 Corpus Compilation

We compiled a corpus of Vedic Sanskrit texts from the TITUS digital text platform (Thesaurus Indogermanischer Text- und Sprachmaterialien). The dataset includes the major Saṁhitā (hymn collections) and Brāhmaṇa (prose commentary) texts of the Vedic corpus, as well as Āraṇyaka and Upaniṣad sections. All the following texts are annotated with the original accent marks. The corpus comprises eight texts:

AVŚ	Atharvaveda Saṁhitā (Śaunaka recension)
MS	Maitrāyaṇī Saṁhitā (Black Yajurveda)
R̥V	R̥gveda Saṁhitā (R̥gveda hymns)
R̥VKh	R̥gveda Khilāni (R̥gveda appendix hymns)
ŚBM	Śatapatha Brāhmaṇa (Mādhyandina recension, a brāhmaṇa of VS)
TB	Taittirīya Brāhmaṇa (a brāhmaṇa of TS)
TS	Taittirīya Saṁhitā (Black Yajurveda)
VS	Vājasaneyi Saṁhitā (White Yajurveda)

These texts span three Vedas (R̥gveda, Atharvaveda and Yajurveda) and represent comprehensive coverage of Vedic genres. Each text in our corpus is provided in transliterated form with diacritical marks following the ISO 15919 standard, which allows encoding Vedic accent as acute (◌́) and grave (◌̀) marks on vowels. As accent marking practices in Devanagari differ substantially across literatures, we adopted the Latin transliteration with diacritics to ensure consistency.

We obtained the texts in accented form from TITUS, which has digitized scholarly editions of these works (e.g., the Atharvaveda Śaunaka edition by Roth & Whitney 1856, etc., as curated in TITUS). We then removed all accent notation from the corpus to create training inputs, with the original accented versions serving as reference outputs.

The dataset was split into training, validation, and test sets in an 8:1:1 ratio by random partitioning at the verse or sentence level that has two or more words, ensuring that there is no overlap of exact verses across sets.

4.2 Dataset Statistics

The whole dataset consists of 108,076 text samples. Each sample is relatively short, with an average length of about six words (mean = 6.03, standard deviation = 5.72). Most texts contain between three and seven words, while the longest example reaches 148 words. The distribution of text lengths for the training, validation, and test sets is visualized in box plots (see Figure 1), illustrating that the overall length distribution remains consistent across subsets.

In terms of vocabulary, the dataset contains a total of 651,337 space-delimited "words" and 133,873 unique "word forms". However, because

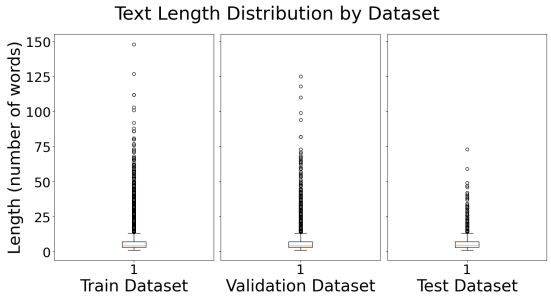

Figure 1: Box plots showing the distribution of text lengths (in words) across training, validation, and test sets.

sound changes called sandhi can combine multiple underlying word forms into a single surface form, the true number of lexical words is likely higher than these counts suggest. The lexical density, defined as the ratio of unique words to total tokens, is 0.2055, which indicates a moderate level of lexical diversity. On average, each text contains approximately six words.

Accent annotations were also analyzed. The average number of accents per text is 5.53 (standard deviation = 6.18), with values ranging from 0 to 154. The median is four accents per text, suggesting that most utterances include a small number of accented segments. Figure 1 shows the distribution of accent counts across the dataset.

Overall, these statistics demonstrate that the corpus is composed primarily of short, lexically varied utterances, with accent patterns distributed broadly but skewed toward lower counts.

Our dataset is publicly available at: https://huggingface.co/datasets/yzk/vedic-accent-restoration-dataset.

5 Models and Fine-Tuning

We fine-tuned two proprietary large language models and two open-weight models on the accent restoration task.

The first model is a Llama 3.1 8B Instruct model (Grattafiori et al., 2024), an eight-billion-parameter instruction-tuned language model from the Llama series (Meta AI). We applied LoRA (Low-Rank Adaptation) (Hu et al., 2022) to fine-tune this model efficiently. We set the LoRA rank to 16 and fine-tuned only the query and value projection matrices of each transformer layer, with all other weights kept fixed. The training objective was a straightforward sequence-to-sequence generation: the model takes an unaccented Vedic text sequence as input and is trained to output the same sequence with correct accent marks inserted in the appropriate positions. We fine-tuned for 10 epochs (approximately 50k update steps) with a learning rate of 2e-4, using the AdamW optimizer. The model converged quickly, likely due to the simplicity of the output (accent markers) relative to the rich pretraining of the Llama model.

The second model is OpenAI GPT-4.1 nano (OpenAI, 2025), a proprietary LLM accessible via API. This model is an instruction-following variant of GPT-4 with a smaller parameter scale. We performed supervised fine-tuning (SFT) on GPT-4.1 nano by supplying our training pairs through the OpenAI fine-tuning API. The model was fine-tuned in a similar sequence-to-sequence fashion: each training example was presented as a prompt consisting of an unaccented Vedic sentence, with the expected accented sentence as the completion. We fine-tuned GPT-4.1 nano for one epoch over the training data (the maximum allowed by OpenAI's guidelines for this model). Despite the model's smaller size compared with full GPT-4, it benefits from GPT-4's advanced initialization and instruction tuning. We anticipated that GPT-4.1 nano might capture accent patterns from context even without seeing as many examples, due to its strong zero-shot capabilities.

The third proprietary model is Google Gemini 2.5 Flash (LLC, 2025), a fast and instruction-optimized variant of the Gemini series. We fine-tuned this model using the Gemini API, following Google's official fine-tuning guidelines. To align with these recommendations, we limited the training dataset size by randomly sampling 2,000 sentence pairs from our full dataset. The fine-tuning procedure followed the same supervised sequence-to-sequence format as with GPT-4.1 nano: the input was an unaccented sentence and the output the correctly accented version. Although the smaller training size constrained exposure, the model adapted efficiently and demonstrated strong contextual generalization, suggesting that Gemini's robust instruction tuning and multilingual pretraining provide useful inductive bias for accent restoration tasks.

6 Evaluation Setup

We evaluate the models on the held-out test set of Vedic sentences/verses with gold-standard accent markings. The primary evaluation metrics are Pre-

cision, Recall, and F1-score for accent restoration, computed at the character level on vowels. An accent prediction is considered correct if the model outputs the correct diacritic (e.g., an acute accent) on the exact vowel that is accented in the reference. Precision thus reflects the fraction of accent marks inserted by the model that are correct, while Recall reflects the fraction of actual reference accent marks that the model successfully restored. F1 is the harmonic mean of Precision and Recall, summarizing overall accuracy of accent placement.

In addition to character-level metrics, we also examine CER (character error rate), WER (word error rate), and ChrF1 (character-level F1 score) (Popović, 2015), to provide a more holistic view of model performance.

The evaluation was performed separately for each model. We used the same test set for all models, containing around 2,000 lines covering all included texts. This ensures a fair comparison under identical conditions. No post-processing was applied to the model outputs; we compare raw model output to the reference after normalizing Unicode combining characters for fairness.

7 Results

7.1 Overall Performance

All fine-tuned models substantially outperform their pre-trained baselines across all metrics. The Llama 3.1 8B model after supervised fine-tuning achieves the best overall performance, with a precision of 0.916, recall of 0.841 and F1-score of 0.877. Its word error rate (WER) is the lowest among the tested models, and it achieves the highest ChrF1 score (87.5). Although its character error rate (CER) is not the absolute minimum, it remains competitive.

GPT-4.1 nano and Gemini 2.5 models also show strong gains after fine-tuning, indicating that SFT effectively adapts each base model to the specific linguistic task of accent restoration. In particular, GPT-4.1 nano's CER of 0.062 suggests it produces fewer local character-level errors, while Gemini 2.5 Flash maintains balanced precision and recall, leading to a stable F1 of 0.780. These proprietary models already achieve strong performance even before SFT.

7.2 Error Analysis

A common error type observed across models is over-generation of accents (false positives), where an accent mark is added to an unaccented syllable. Such cases often occur adjacent to the correct position. Missed accents (false negatives) are typically found in long compounds or phrases. These patterns suggest that local contextual cues play a central role in the models' predictions.

Overall, these results demonstrate that fine-tuned large language models are capable of restoring complex Vedic accent patterns with high accuracy, capturing both surface orthographic and deeper phonological regularities. The open-weight LoRA-tuned Llama 3.1 8B model achieves performance comparable to the proprietary GPT-4.1 nano model while requiring significantly less computational cost, making it an attractive option for deployment in Sanskrit text processing pipelines.

7.3 Improvement Rates by Text Type

To examine whether fine-tuning effects differ across textual genres, we computed improvement rates for each corpus, Ṛgveda (ṚV), Yajurveda (YV), and Atharvaveda (AV), based on the improvement from pre-trained to fine-tuned models.

Table 2 summarizes the relative improvements for core metrics.

Overall, the improvement trends are broadly consistent across the three Vedic corpora. All show large reductions in character and word error rates (ranging from roughly 50% to 130% decreases), and substantial increases in precision and overall F1-scores. Although the exact magnitudes vary slightly with the largest CER reduction observed in the ṚV and the strongest gain in ChrF1 in the AV, the general pattern suggests that fine-tuning improves performance in a relatively uniform way across different Vedic text types.

The modest differences (within about 10–15% across corpora) imply that the model's learning is not strongly biased toward a specific Vedic text. This indicates that the fine-tuned model captures accentual patterns that generalize well across textual traditions, rather than overfitting to any single recension or genre.

7.4 Improvement Rates by Text Category

We also compared improvement rates between the **Saṁhitā** and non-**Saṁhitā** (Brāhmaṇa, Āraṇyaka, Upaniṣad) groups to investigate whether the prose or metrical style of the text affects restoration accuracy. For simplicity, the Black Yajurveda, which traditionally contains both Saṁhitā and Brāhmaṇa portions, was counted as part of the Saṁhitā group.

Model	Precision↑	Recall↑	F1↑	CER↓	WER↓	ChrF1↑
GPT-4.1 nano (Before SFT)	0.609	0.020	0.039	0.288	0.858	45.6
GPT-4.1 nano (After SFT)	0.752	0.676	0.712	**0.062**	0.322	79.6
Gemini 2.5 Flash(Before SFT)	0.551	0.191	0.284	0.698	0.863	22.6
Gemini 2.5 Flash (After SFT)	0.789	0.771	0.780	0.109	0.249	83.5
Llama 3.1 8B (Before SFT)	0.452	0.034	0.064	0.249	0.894	48.1
Llama 3.1 8B (After SFT)	**0.916**	**0.841**	**0.877**	0.096	**0.161**	**87.5**

Table 1: Evaluation results on Vedic accent restoration. Bold values indicate the best performance for each metric.

Metric	RV	YV	AV
CER (%)	131.66	75.78	80.56
WER (%)	74.78	61.16	52.58
ChrF1 (%)	60.61	59.17	67.81
Precision (%)	54.40	63.80	53.08
Recall (%)	32.00	11.10	19.01
F1 (%)	20.41	26.25	34.64

Table 2: Relative improvement rates by text type.

Similarly, Brāhmaṇa texts often include direct quotations from the Saṁhitā, but these were not separated out and were counted within the Brāhmaṇa group.

Metric	Saṁhitā	Non-Saṁhitā
CER (%)	84.72	79.35
WER (%)	62.13	58.20
ChrF1 (%)	63.41	60.02
Precision (%)	57.92	54.10
Recall (%)	8.43	5.26
F1 (%)	28.46	26.89

Table 3: Improvement rates by text category.

As shown in Table 3, the improvement rates for both groups are comparable across all metrics. The Saṁhitā group shows slightly higher reductions in character and word error rates (around 80-85%), but the differences from the Brāhmaṇa group remain within a narrow range of 3-5%. This suggests that fine-tuning improved model performance in a balanced manner, regardless of textual genre or prosodic complexity.

The result further indicates that the model generalizes well across metrical and prose texts alike, capturing accent patterns that apply uniformly to both verse and explanatory prose. Given the mixed nature of Vedic textual traditions and the presence of quotations across sections, such genre-independent gains are a desirable property for robust automatic accent restoration.

8 Conclusion

We presented a study on restoring Vedic Sanskrit accent marks with fine-tuned large language models, achieving 87.7% F1 on inserting correct accentual markings into unaccented texts. Beyond surface accuracy, this performance suggests that the model has internalized core regularities of Vedic phonology and morphosyntax, learning not just where accents occur, but also why they occur, as accent placement in Vedic reflects clause structure, lexical accent and sandhi outcomes.

This capability opens concrete avenues for downstream Vedic NLP. Accented input can sharpen sandhi splitting and morphological disambiguation and provide informative signals for syntactic parsing and machine translation. In the broader Sanskrit pipeline, accent restoration can serve as a front-end normalization step that improves robustness in (i) post-OCR correction (Nehrdich et al., 2024; Maheshwari et al., 2022), (ii) Vedic OCR workflows (Tsukagoshi et al., 2025), (iii) compound type identification (Krishnan et al., 2025), and (iv) Sanskrit translation systems (Pandey et al., 2022; Punia et al., 2020). In each case, accent cues provide linguistically grounded features that downstream models can exploit.

Future work will scale to larger base models and explore multitask and pipeline training, e.g., joint learning with parsing or translation, or end-to-end systems that perform OCR, accent restoration and then analysis. We also plan to test portability to other historical languages that use diacritical systems. Ultimately, restoring Vedic accents is not an orthographic nicety; it is a means to recover latent

linguistic information and to enhance the fidelity of subsequent language processing tasks.

Limitations

Our study focuses exclusively on the task of Vedic accent restoratioin, and we do not empirically evaluate the impact of the task on downstream NLP tasks such as sandhi splitting, morphological analysis, syntactic parsing, or machine translation. While linguistic theory suggests that explicit phonoclogical marking may be beneficial, confirming these effects requires further systematic evaluation.

In addition, our experiments rely on a limited set of textual source, which do not fully represent the diversity of Vedic textual traditions, recensions, or orthographic conventions.

Another limitation concerns the evaluation of accent placement in compound nouns. In Vedic Sanskrit, compounds represent a challenging case for accent restoration (section 2). Ideally, we should evaluate the models on such minimal pairs. However, our current test set does not contain representative examples of these compounds, in part because we did not manually curate this subset when constructing the splits. A future version of the dataset should incorporate a balanced selection of accentually contrastive compounds, enabling a more systematic evaluation of model performance on accent-based semantic and morphological distinctions.

Acknowledgments

This work was supported by The Nippon Foundation HUMAI Program.

References

Zeina Aldallal, Sara Chrouf, Khalil Hennara, Mohamed Motaism Hamed, Muhammad Hreden, and Safwan AlModhayan. 2025. Sadeed: Advancing arabic diacritization through small language model. *Preprint*, arXiv:2504.21635.

Yannis Assael, Thea Sommerschield, and Jonathan Prag. 2019. Restoring ancient text using deep learning: a case study on Greek epigraphy. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6368–6375, Hong Kong, China. Association for Computational Linguistics.

Trung Duc Anh Dang and Thi Thu Trang Nguyen. 2020. TDP – a hybrid diacritic restoration with transformer decoder. In *Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation*, pages 76–83, Hanoi, Vietnam. Association for Computational Linguistics.

Toshifumi Gotō. 2013. *Old Indo-Aryan morphology and its Indo-Iranian background*. Number 849. Bd. in Sitzungsberichte / Österreichische Akademie der Wissenschaften, Philosophisch-Historische Klasse. Verlag der Österreichischen Akademie der Wissenschaften.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Sriram Krishnan, Pavankumar Satuluri, Amruta Barbadikar, T S Prasanna Venkatesh, and Amba Kulkarni. 2025. Compound type identification in Sanskrit. In *Computational Sanskrit and Digital Humanities - World Sanskrit Conference 2025*, pages 90–108, Kathmandu, Nepal. Association for Computational Lingustics.

Google LLC. 2025. Gemini models - gemini 2.5 flash. `https://ai.google.dev/gemini-api/docs/models#gemini-2.5-flash`. Accessed: 2025-11-24.

Arthur Anthony Macdonell. 1910. *Vedic Grammar*. Karl J. Trübner.

Ayush Maheshwari, Nikhil Singh, Amrith Krishna, and Ganesh Ramakrishnan. 2022. A benchmark and dataset for post-OCR text correction in Sanskrit. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6258–6265, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

OpenAI. 2025. Introducing gpt-4.1 in the api. `https://openai.com/index/gpt-4-1/`. Accessed: 2025-11-24.

Mrinal Pandey, Rashmikiran Pandey, and Alexey Nazarov. 2022. Machine translation of vedic sanskrit using deep learning algorithm. In *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, pages 1477–1480.

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.

Ravneet Punia, Aditya Sharma, Sarthak Pruthi, and Minni Jain. 2020. Improving neural machine translation for Sanskrit-English. In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, pages 234–238, Indian Institute of Technology Patna, Patna, India. NLP Association of India (NLPAI).

Ryan Sandell. 2024. Preserving computational uniformity in vedic sanskrit stress assignment. Abstract for workshop "Nonuniformity in Morphophonology across Frameworks", ERSaF / Arndt-Lappe Project, Trier. PDF available online.

Yuzuki Tsukagoshi, Ryo Kuroiwa, and Ikki Ohmukai. 2025. Towards accent-aware Vedic Sanskrit optical character recognition based on transformer models. In *Computational Sanskrit and Digital Humanities - World Sanskrit Conference 2025*, pages 70–80, Kathmandu, Nepal. Association for Computational Linguistics.

David Yarowsky. 1994. Decision lists for lexical ambiguity resolution: Application to accent restoration in Spanish and French. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 88–95, Las Cruces, New Mexico, USA. Association for Computational Linguistics.

A Training Details

A.1 Training Configurations

The training configurations used for the Llama 3.1 8B Instruct, GPT-4.1 nano, and Gemini 2.5 Flash models are summarized below.

Llama 3.1 8B Instruct

- LoRA rank: 16
- LoRA alpha: 16
- LoRA dropout: 0.0
- Learning rate: 3e-4
- Learning rate scheduler: linear
- Warmup steps: 10
- Weight decay: 0.01
- Epochs: 10
- Batch size: 4
- Gradient accumulation steps: 8
- Optimizer: AdamW

GPT-4.1 nano

- Epochs: 1 (default)
- Batch size: 32
- Learning rate multiplier: 0.1

Gemini 2.5 Flash

- Epochs: 22 (automatically determined)
- Adapter size: 4 (default)

A.2 Training Data Format

For all models, the training data was formatted as pairs of input-output sequences. The input sequence consisted of the unaccented Vedic text, while the output sequence contained the same text with correct accent marks inserted.

```
Please restore the Vedic accents in the
following Vedic Sanskrit text.

### Input:
{input_text}

### Target:
{output_text}
```

Dataset contains the source, target and text_id pairs in JSONL format as follows:

```
  {
  "text_id": "YVB_MS_2_3_4_ai",
  "source": "tenāyuṣāyuṣmān edhi",
  "target": "ténāyuṣāyuṣmān edhi"
}
...
```