Accent Placement Models for Rigvedic Sanskrit Text

Akhil Rajeev P and Annarao Kulkarni
Indian Heritage Language Computing Team
Special and Strategic Projects (SSP) Group
Centre for Development of Advanced Computing (C DAC), Bangalore
akhilrajeev@cdac.in kulkarni@cdac.in

Abstract

The Rigveda, among the oldest Indian texts in Vedic Sanskrit, employs a distinctive pitch-accent system : udātta, anudātta, svarita whose marks encode melodic and interpretive cues but are often absent from modern e-texts. This work develops a parallel corpus of accented-unaccented ślokas and conducts a controlled comparison of three strategies for automatic accent placement in Rigvedic verse: (i) full fine-tuning of ByT5, a byte-level Transformer that operates directly on Unicode combining marks, (ii) a from-scratch BiLSTM-CRF sequence-labeling baseline, and (iii) LoRA-based parameter-efficient fine-tuning atop ByT5.

Evaluation uses Word Error Rate (WER) and Character Error Rate (CER) for orthographic fidelity, plus a task-specific Diacritic Error Rate (DER) that isolates accent edits. Full ByT5 fine-tuning attains the lowest error across all metrics; LoRA offers strong efficiency-accuracy trade-offs, and BiLSTM-CRF serves as a transparent baseline. The study underscores practical requirements for accent restoration - Unicode-safe preprocessing, mark-aware tokenization, and evaluation that separates grapheme from accent errors - and positions heritage-language technology as an emerging NLP area connecting computational modeling with philological and pedagogical aims. Results establish reproducible baselines for Rigvedic accent restoration and provide guidance for downstream tasks such as accent-aware OCR, ASR/chant synthesis, and digital scholarship.

1 Introduction

The *Rigveda*, an ancient collection of *ṛk*-s (hymns) composed in Vedic Sanskrit, encodes its recitational tradition through a sophisticated accent system. Words in Vedic Sanskrit bear accented syllables, and each Veda has its own set of accent markers, with some shared across Vedic corpora. The detailed list of accent markers used in the Vedas is standardized in ISO/ISCII_Ammex-G.[1]

Rigvedic phonology distinguishes three tones: **Udātta** (high tone, normally unmarked), **Anudātta** (low tone, shown by a mark below the character; U+0952), and **Svarita** (rising-falling tone, marked above the character; U+0951). These accent signs guide chanting and preserve tonal precision in oral tradition.

These markers prescribe tone or pitch for recitation, and are also embedded semantic units: a change in accent can alter the meaning of a word. The phonetic rules governing accents are described in the *Prātiśākhya*-s and *Śikṣāśāstra* texts, with the *Ṛgveda Prātiśākhya* serving as the authoritative source for Rigvedic phonology. The *Nighaṇṭu* provides a lexicon of Vedic words, and fully appreciating why a syllable bears a particular accent typically requires expertise in Sanskrit grammar, *chandas* (metrics), *nirukta* (etymology), and phonetics.

Despite its linguistic centrality, many searchable e-texts and NLP resources omit accents due to encoding limitations or design choices prioritizing searchability (Unicode Consortium, 2025a,b; Cologne Sanskrit Lexicon, 2020). This omission hampers philological research, chanting pedagogy, and speech systems that depend on tonal cues (Hellwig et al., 2020; Kumar et al., 2025). Accent distinctions are essential for oral instruction, yet learners using unaccented corpora cannot reconstruct melodic contours. In speech technology, ASR or TTS systems trained on unaccented data fail to capture prosody vital for faithful recitation. Automatic accent restoration therefore represents both a **technical challenge**-a low-resource sequence labeling task on metrical Sanskrit verse with pervasive sandhi-and a **cultural-heritage challenge** vital to

[1] For descriptive overviews see Extended Character Set for Vedic IS 13194:1991; for phonetic discussion of the independent *svarita*, see Beguš (2016).

preserving an oral tradition.

Recent NLP advances make such restoration feasible. Byte- and character-level transformers restore diacritic-like markers without brittle tokenization (Xue et al., 2022), and parameter-efficient fine-tuning lowers adaptation cost for niche low-resource domains (Houlsby et al., 2019; Hu et al., 2022). Parallel work on Arabic, Hebrew, and Yorùbá shows the effectiveness of normalization-aware pipelines and diacritic-sensitive architectures (Alqahtani et al., 2020; Gershuni and Pinter, 2022; Rosenthal and Shaked, 2024; Cohen et al., 2024; Olawole et al., 2024), suggesting methodological transferability even though Vedic accent remains unexplored.

This study investigates whether modern AI models can automatically accent unaccented Rigvedic hymns without expert linguistic rules. We construct a parallel corpus of accented-unaccented verse pairs and evaluate three strategies: (i) full ByT5 fine-tuning (Xue et al., 2022), (ii) a BiLSTM-CRF sequence labeler (Huang et al., 2015), and (iii) LoRA-based ByT5 tuning (Hu et al., 2022). Evaluation using Word, Character, and Diacritic Error Rates (WER, CER, DER) shows full ByT5 achieves the lowest errors, LoRA balances accuracy and efficiency, and BiLSTM-CRF provides a reproducible baseline. Our released corpus and code aim to advance accent-aware OCR, ASR, and pedagogy for Vedic studies (Tsukagoshi et al., 2025; Kumar et al., 2025).

2 Related Work

Computational Sanskrit and Vedic resources: Sanskrit NLP has focused on segmentation, sandhi splitting, morphology, and syntax - foundations for accent restoration. Early tools such as SANSKRITTAGGER and sentence boundary detectors processed punctuation-light text (Hellwig, 2010, 2016). Neural methods advanced segmentation via character-CNN/LSTM and graph inference (Hellwig and Nehrdich, 2018; Krishna et al., 2016). The Sanskrit Heritage platform, distributed processing stacks, and the UD-style *Vedic* Treebank supply lexical and syntactic supervision useful for accent diagnostics (Goyal et al., 2012; Huet, 2003–; Hellwig et al., 2020). Indian research further formalized dependency relations for Sanskrit grammar (Kulkarni et al., 2020). Recent work introduces accent-aware OCR and ASR benchmarks for Vedic Devanagari, defining a new context for restoration (Tsukagoshi et al., 2025; Kumar et al., 2025).

Standardization of Vedic characters: The extended character repertoire for Vedic scripts defined in Annex G of Extended Character Set for Vedic IS 13194:1991 (ISCII) provided the initial framework for digital representation of Vedic accents and combining marks. This early standard, based on an 8-bit encoding scheme, served as the foundation for later Unicode integration. Its specifications were incorporated into the Unicode *Vedic Extensions* block (U+1CD0-U+1CFX), ensuring compatibility with Devanagari and related Brahmic scripts and enabling cross-platform rendering of accents and tonal signs. This standardization has been central to developing searchable, accent-preserving corpora and tools for computational Vedic studies (Unicode Consortium, 2025a,b).

Diacritics and accents beyond Sanskrit: Arabic, Hebrew, and Yorùbá diacritic restoration offer methodological parallels. Accuracy correlates with modeling capacity and domain adaptation, from multitask setups to "diacritics-in-the-wild" corpora (Alqahtani et al., 2020; Elgamal et al., 2024). Hebrew and Yorùbá studies use compact character LSTMs or transformer variants (NAKDIMON, MenakBERT, D-Nikud, T5) (Gershuni and Pinter, 2022; Cohen et al., 2024; Rosenthal and Shaked, 2024; Olawole et al., 2024).

Modeling choices: Byte and character-level transformers avoid fragile tokenization in diacritic-rich scripts; the T5 variant used here is ByT5-Sanskrit ((Nehrdich et al., 2024)), a byte-level model trained specifically for Sanskrit NLP tasks, which avoids fragile tokenization in diacritic-rich scripts and performs strongly on UTF-8 text. Parameter-efficient transfer (adapters, LoRA) lowers cost for low-resource tasks such as Rigvedic accenting (Houlsby et al., 2019; Hu et al., 2022). BiLSTM-CRF remains a transparent baseline for sequence labeling (Huang et al., 2015).

3 Dataset

An in-house, validated *Rigveda* corpus developed at C-DAC is used, comprising 10,552 hymns organized into 10 *maṇḍalas* and 1,028 *sūktas*. From this resource, a parallel corpus of 22,740 aligned verse pairs is constructed: each entry pairs an unaccented verse with its diacritically marked counterpart for

supervised training and evaluation.[2]
Provenance fields (maṇḍala-sūkta-ṛc identifiers) accompany each record.
Example:
Unaccented

अग्निमीळे पुरोहितं यज्ञस्य देवमृत्विजम्।

Accented

अ॒ग्निमी॑ळे पु॒रोहि॑तं य॒ज्ञस्य॑ दे॒वमृ॒त्विज॑म्।

The accented form adds pitch cues via combining marks while core graphemes remain unchanged - motivating CER for orthography and DER for accent-specific edits.

Splits: Train / test / dev (validation) partitions are drawn from the in-house Rigveda corpus, with stratification by maṇḍala and sūkta length to mitigate topical leakage. From the total of 22,740 aligned verse pairs, we adopt the train / validation / test split, as shown in Table 1.

Table 1: **Train / development / test split of the aligned corpus (22,740 verse pairs).**

Split	Verse pairs	Percentage
Train	**19,329**	**85%**
Development	**2,274**	**10%**
Test	**1,137**	**5%**
Total	**22,740**	**100%**

4 Methodology

We evaluate three models:

1. **Full Fine-tuning (ByT5):** The multilingual ByT5 model was fine-tuned end-to-end. We used learning rate $3e{-}5$, batch size 32, and trained for 10 epochs.

2. **BiLSTM-CRF:** This model used 256-d embeddings, a 2-layer BiLSTM (hidden size 512), and a CRF decoding layer. Dropout 0.3 was applied. Training used Adam ($lr = 1e{-}3$) for 20 epochs.

3. **LoRA Fine-tuning (ByT5):** LoRA with rank 8 and $\alpha = 16$ was applied to the self-attention projection matrices. Only 0.5% of parameters were updated.

[2]The C-DAC Rigveda parallel corpus will be made available through the Indian Knowledge Base platform at https://indianknowledgebase.in/.

5 Evaluation Metrics

System performance is assessed using three complementary string-level metrics designed to capture lexical, orthographic, and diacritic-specific accuracy.

- **Word Error Rate (WER)** measures **token-level edit distance** (Insertions, Deletions, Substitutions), reflecting overall lexical fidelity. It is normalized by the number of reference tokens.

- **Character Error Rate (CER)** computes **character-level edit distance**, excluding whitespace. This metric is sensitive to fine-grained orthographic deviations, making it suitable for morphologically rich Sanskrit text.

- **Diacritic Error Rate (DER)** isolates errors in **accent symbols (diacritics)** alone, disregarding base characters. It quantifies the precision of accent placement (tonal correctness), normalized over the total diacritic instances in the reference.

Together, these metrics provide complementary views of model behavior: WER captures global token accuracy, CER measures character integrity, and DER specifically reflects accent restoration performance at the sub-character level.

6 Results and Discussion

Method	WER	CER	DER
Full FT (ByT5)	**0.1023**	**0.0246**	**0.0685**
BiLSTM–CRF	0.2367	0.0448	0.3197
LoRA FT (ByT5)	0.3614	0.1042	0.1598

Table 2: Performance of Sanskrit accent placement models. Best scores in bold.

Our findings highlight three insights:

Transformer advantage: Full ByT5 fine-tuning outperforms alternatives by a wide margin, confirming that large pretrained models adapt well even to heritage tasks with limited training data.

Diacritic modeling challenge: BiLSTM–CRF achieves tolerable WER and CER but fails dramatically in DER. This suggests that traditional models cannot capture pitch diacritic patterns without explicit linguistic priors.

Efficiency vs. fidelity: LoRA reduces trainable parameters by orders of magnitude but suffers in WER/CER. Interestingly, its DER surpasses BiLSTM–CRF, indicating that localized diacritic learning may be partially preserved.
Beyond metrics, these results matter for applications: chanting synthesis requires low DER, while digital philology may tolerate higher CER if semantic accents are preserved.

7 Error Analysis

To examine model behavior beyond aggregate metrics, we manually analyzed 200 mispredicted verses from the test set, comparing error tendencies across ByT5 (full fine-tuning), BiLSTM–CRF, and ByT5-LoRA. Four categories emerged: accent misplacement, omission or over-generation, accent-type confusion, and boundary errors.

Accent Misplacement :The dominant category (46.8%) involved accents shifted by one mora within the correct syllabic span (e.g., *devam̱ṛ̱tvijam* → *devamṛ̱tvijam*). ByT5 had the lowest misplacement rate (18.2%), while BiLSTM–CRF (41.5%) and LoRA (33.4%) showed weaker morphemic control, suggesting that ByT5's byte-level encoding captures compound co-occurrence patterns, whereas LoRA underfits longer phonological sequences.

Omission and Over-generation: These formed 26.3% of all errors, mostly in verses with multiple enclitic particles (*ha, ca, u*). BiLSTM–CRF tended to over-generate (14.8%), while LoRA favored omission (11.2%), reflecting a conservative decoding bias from low-rank adaptation.

Accent-Type Confusion: About 15.1%) involved udātta–svarita swaps, common in reduplicated or rhythmic verb forms, indicating a need for explicit tone hierarchy modeling.

Boundary and Tokenization Errors: A smaller share (8.7%)) arose from accent drift across pāda or punctuation boundaries. BiLSTM–CRF was most affected due to fixed segmentation, whereas ByT5's byte-level representation mitigated drift.

Cross-Metric Correlation: Diacritic Error Rate (DER) correlated strongly with Character Error Rate (CER) ($r = 0.82$) but weakly with Word Error Rate (WER) ($r = 0.39$), confirming accent restoration as a sub-character orthographic task.

Qualitative Observations: Mid-frequency stems (*agní*, *soma*, *indra*) were accented correctly across models, while rare words like *puruṣṭuta* showed erratic realizations.

8 Conclusion

We introduced the first benchmark for automatic accent restoration in Rigvedic Sanskrit, evaluated with *Word Error Rate (WER)*, *Character Error Rate (CER)*, and the task-specific *Diacritic Error Rate (DER)* focused on accent deviations. Full fine-tuning of ByT5 achieves the strongest results, LoRA balances efficiency and accuracy, and BiLSTM–CRF serves as a transparent baseline. This work demonstrates that modern NLP methods can be effectively adapted to heritage-language processing despite data sparsity and domain constraints. Beyond restoration accuracy, the framework holds promise for enabling systematic *prosodic annotation* of Vedic corpora, thereby facilitating deeper linguistic and chanting analyses. Its interpretability could further support *explainable AI* approaches to modeling oral–textual correspondences.

9 Acknowledgments

We acknowledge Dr. Janaki C. H. and Dr. S. D. Sudarsan for their guidance and administrative facilitation throughout this work. Additionally,our sincere thanks go to Sinchana Bhat and Archana Bhat, Classical Sanskrit Linguists at C-DAC Bangalore, for their crucial contributions in data annotation and human evaluation.

10 Limitations

Our study has the following limitations:

1. **Data size:** The corpus is relatively small compared to modern NLP benchmarks, restricting model generalization and robustness.

2. **Evaluation metrics:** We use WER, CER, and DER, which measure surface accuracy but do not capture alignment with deeper metrical or phonological rules described in traditional sources.

3. **Model coverage:** We evaluate three approaches (ByT5, LoRA, BiLSTM–CRF). Other architectures, such as non-autoregressive transformers, graph-based methods, or phonology-aware encoders, are not explored.

References

Sawsan Alqahtani, Mohammed Attia, Kareem Darwish, Hamdy Mubarak, Ahmed Abdelali, and Mona Diab. 2020. A multitask learning approach for diacritic restoration. In *Proceedings of ACL 2020*, pages 8238–8247.

Gašper Beguš. 2016. The phonetics of the independent svarita in vedic. In *Proceedings of the 26th Annual UCLA Indo-European Conference*.

Ido Cohen, Jacob Gidron, and Idan Pinto. 2024. Menakbert – hebrew diacriticizer. *arXiv preprint arXiv:2410.02417*.

Cologne Sanskrit Lexicon. 2020. Vedic accent encoding in the cologne sanskrit lexicon.

Salman Elgamal, Ossama Obeid, Nizar Habash, Go Inoue, and 1 others. 2024. Arabic diacritics in the wild: Exploiting opportunities for maximal diacritization. In *Proceedings of ACL 2024*.

Extended Character Set for Vedic IS 13194:1991. —. Iscii annex-g: Vedic accent markers (vedic extensions to iscii / iso). `https://www.vedavid.org/vedic_ISO/ISCII_Annex-G.pdf`. Accessed: December 1, 2025.

Elazar Gershuni and Yuval Pinter. 2022. Restoring hebrew diacritics without a dictionary. In *Findings of NAACL 2022*, pages 1012–1022.

Pawan Goyal, Gérard Huet, Amba Kulkarni, Peter Scharf, and Ralph Bunker. 2012. A distributed platform for sanskrit processing. In *Proceedings of COLING 2012*.

Oliver Hellwig. 2010. Sanskrittagger: A stochastic lexical and pos tagger for sanskrit. In *Proceedings of LREC 2010*.

Oliver Hellwig. 2016. Detecting sentence boundaries in sanskrit texts. In *Proceedings of COLING 2016*, pages 288–297.

Oliver Hellwig and Sebastian Nehrdich. 2018. Sanskrit word segmentation using character-level recurrent and convolutional neural networks. In *Proceedings of EMNLP 2018*, pages 4688–4697.

Oliver Hellwig, Salvatore Scarlata, Elia Ackermann, and Paul Widmer. 2020. The treebank of vedic sanskrit. In *Proceedings of LREC 2020*, pages 5137–5146.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, pages 2790–2799. PMLR.

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 (ICLR)*.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Gérard Huet. 2003–. The sanskrit heritage site.

Amrith Krishna, Pavankumar Satuluri, and Pawan Goyal. 2016. Word segmentation in sanskrit using path constrained random walks. In *Proceedings of COLING 2016*, pages 494–504.

Amba Kulkarni, Pavankumar Satuluri, Sanjeev Panchal, Malay Maity, and Amruta Malvade. 2020. Dependency relations for sanskrit parsing and treebank. In *Proceedings of the 19th Workshop on Treebanks and Linguistic Theories*, pages 125–134.

Sujeet Kumar, Pretam Ray, Abhinay Beerukuri, Shrey Kamoji, Manoj Balaji Jagadeeshan, and Pawan Goyal. 2025. Vedavani: A benchmark corpus for asr on vedic sanskrit poetry. In *Computational Sanskrit and Digital Humanities, World Sanskrit Conference*, pages 81–89, Kathmandu, Nepal. ACL.

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.

Akindele Michael Olawole, Jesujoba O. Alabi, Aderonke Busayo Sakpere, and David Ifeoluwa Adelani. 2024. Yad: Leveraging t5 for improved automatic diacritization of yorùbá text. In *AfricaNLP Workshop @ ICLR*.

Adi Rosenthal and Nadav Shaked. 2024. D-nikud: Enhancing hebrew diacritization with lstm and pretrained models. *arXiv preprint arXiv:2402.00075*.

Yoshitomo Tsukagoshi, Nikhil Sharma, and Pawan Goyal. 2025. Towards accent-aware vedic sanskrit optical character recognition. In *Computational Sanskrit and Digital Humanities, World Sanskrit Conference*, pages 71–80, Kathmandu, Nepal. ACL.

Unicode Consortium. 2025a. The unicode standard, version 17.0: Devanagari code chart (u+0900–u+097f). Code charts.

Unicode Consortium. 2025b. The unicode standard, version 17.0: Vedic extensions (u+1cd0–u+1cff). Code charts.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Ankur Barua, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. In *Transactions of the Association for Computational Linguistics*, volume 10, pages 2916–2931. MIT Press.