

Arthasetu - Neural Machine Translation for Classical Jain Philosophical Text with Cross-System Evaluation

Tamanna Jain

ertamannajain@gmail.com

Maharishi Markandeshwar Engineering College

Neeraj Raheja

Maharishi Markandeshwar Engineering College

Research Article

Keywords: Sanskrit-Hindi machine translation, low-resource NLP, neural machine translation, sequence-to-sequence attention models, domain adaptation, BiLSTM, Jain philosophical text, evaluation methodology

Posted Date: October 15th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10974349/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.
[Read Full License](#)

Additional Declarations: No competing interests reported.

Arthasetu - Neural Machine Translation for Classical Jain Philosophical Text with Cross-System Evaluation

Tamanna Jain¹, Neeraj Raheja²

^{1,2}*Department of Computer Science and Engineering, Maharishi Markandeshwar Engineering College, Mullana, Ambala, Haryana, India*

¹*Corresponding Author: ertamannajain@gmail.com*

²*Corresponding Author: neeraj_raheja2003@mmumullana.org*

Abstract

Sanskrit-Hindi machine translation (MT) remains an underexplored low-resource task, and published prior work has predominantly evaluated on conversational (Laukik) Sanskrit rather than dense classical philosophical text. This paper presents Arthasetu, a Bidirectional LSTM encoder-decoder architecture with Bahdanau attention, developed and evaluated specifically for the classical Jain philosophical Sanskrit-to-Hindi translation sub-domain. We construct and progressively refine three training corpora of increasing domain purity: an original 9,878-sentence classical Jain corpus, an 89,821-sentence corpus augmented with a cross-domain contemporary/encyclopedic source, and a 15,087-sentence corpus expanded exclusively with additional domain-consistent Jain philosophical and devotional texts. We show that cross-domain augmentation improves aggregate metrics while degrading domain-specific performance (Jain-subset BLEU: 1.35), whereas domain-consistent augmentation improves domain-specific performance directly (BLEU: 2.05, chrF++: 13.59, BERTScore F1: 0.686). We further conduct a systematic ablation of tokenization strategy (word-level vs. SentencePiece BPE) and decoding method (greedy vs. beam search), and report a comparative evaluation against Google Translate and NLLB-200 on an identical held-out test set, finding that Arthasetu outperforms NLLB-200 on BLEU (2.05 vs. 0.38) despite NLLB's substantially larger multilingual pretraining. We additionally document and correct a data-leakage defect discovered during development (an evaluation-time training-corpus lookup shortcut inflating BLEU to approximately 99.8), underscoring evaluation-methodology risks specific to small-corpus, low-resource MT research. We conclude with a systematic audit of Sanskrit-language support across major open multilingual MT systems and a proposed hybrid retrieval-generation architecture for production deployment.

Index Terms—*Sanskrit-Hindi machine translation, low-resource NLP, neural machine translation, sequence-to-sequence attention models, domain adaptation, BiLSTM, Jain philosophical text, evaluation methodology.*

1. Introduction

Sanskrit occupies a distinctive position among the scheduled languages of India: it is the liturgical and philosophical language of a substantial body of classical literature — including Jain, Buddhist, and Hindu philosophical, devotional, and cosmological texts — yet it remains severely underrepresented in modern natural language processing resources relative to its cultural significance. Existing machine translation research for the Sanskrit-Hindi pair has largely evaluated systems on conversational (Laukik) Sanskrit corpora, which are lexically and syntactically closer to Hindi and consequently present a substantially easier translation task than dense, compound-heavy classical philosophical Sanskrit.

This work is motivated by a specific, practical objective: the development of a translation system capable of accurately rendering classical Jain philosophical Sanskrit — exemplified by the Tattvartha Sutra and related commentarial and devotional literature — into Hindi, ultimately in service of a conversational assistant for Jain philosophical study. This objective necessitates confronting three challenges largely absent from prior Sanskrit-Hindi MT literature: (i) severe data scarcity specific to a philosophical sub-domain rather than Sanskrit broadly; (ii) a high density of technical, doctrine-specific compound terminology; and (iii) the need for an evaluation methodology robust to the data-leakage risks inherent to small, hand-curated parallel corpora.

The contributions of this paper are as follows:

- 1) We construct and publicly document three Sanskrit-Hindi parallel corpora of increasing scale and varying domain composition, and empirically demonstrate that cross-domain corpus augmentation improves aggregate translation metrics while degrading domain-specific performance — motivating domain-consistent augmentation as the preferable strategy for specialized-domain MT.
- 2) We conduct a systematic ablation across tokenization strategy (word-level vs. subword/BPE) and decoding strategy (greedy vs. beam search), reporting a counterintuitive finding that subword tokenization does not improve performance at the corpus scales evaluated.
- 3) We report a comparative evaluation against Google Translate and NLLB-200 on an identical held-out test set, and separately document a systematic audit of Sanskrit-language support across major open multilingual MT systems.
- 4) We identify and correct a data-leakage defect in an early evaluation pipeline, and discuss its implications for evaluation methodology in low-resource MT research generally.
- 5) We propose a hybrid retrieval-generation system architecture for production deployment, distinguishing this deployment-layer design from the academic evaluation methodology used to report model performance.

2. Related Work

A. Sanskrit-Hindi Machine Translation: Prior Approaches

Pandey and Jha [1] presented an early error analysis of Sahit, a statistical Sanskrit-Hindi translation system, establishing baseline expectations for phrase-based statistical approaches on this language pair. Sethi, Dev, and Bansal [2] proposed a GRU-based encoder-decoder architecture with attention, reporting a BLEU score of 53.8%, compared to 34.56% for a statistical (Moses-based) baseline. A transformer-based approach [3] extended this line of work using a multi-head attention architecture over a corpus exceeding 50,000 parallel sentences, reporting a BLEU score of 67.8%. Bakarola and Nasriwala [4] reported an attention-based sequence-to-sequence approach evaluated on a corpus of approximately 10,000 sentence pairs, reporting a BLEU score as high as 92% with an accompanying 88% human-evaluation accuracy score.

The BLEU scores reported in [2]–[4] are substantially higher than those obtained in the present study (Section 6). Close inspection of published sample translations from [4] indicates that its underlying corpus consists predominantly of Laukik (conversational) Sanskrit — short, everyday-register sentences describing personal states or simple factual claims — rather than classical or

philosophical Sanskrit. Laukik Sanskrit sentences are typically short, syntactically simple, and lexically closer to Hindi, substantially easing the translation task relative to the dense, compound-heavy classical philosophical text addressed in the present work. We are not aware of prior published Sanskrit-Hindi NMT work specifically targeting the Jain philosophical sub-domain for direct comparison. We additionally note that we were unable to locate publicly released datasets for [2] or [3], precluding independent verification of their reported results or domain composition — a recurring reproducibility gap in low-resource NLP research that this work aims to avoid through full dataset and methodology documentation.

B. Multilingual NMT Systems and Sanskrit Coverage

A separate body of work has produced large-scale multilingual NMT systems covering dozens to hundreds of languages from a single trained model. Motivated by the practical need to identify suitable contrastive baselines, we conducted a systematic audit of Sanskrit-language support across major publicly available multilingual MT systems, summarized in Table 1.

Table 1. SANSKRIT LANGUAGE COVERAGE IN MAJOR MULTILINGUAL MT SYSTEMS

System	Publisher	Total Languages	Sanskrit Support
NLLB-200 [5]	Meta AI	200	Yes (san_Deva)
IndicTrans2 [6]	AI4Bharat	22 (Indic)	Yes (san_Deva)
Google Translate	Google	~130+	Yes (closed-source)
mT5 [7]	Google Research	101	No
mBART-50 [8]	Meta AI	50	No
IndicBART [9]	AI4Bharat	11 (Indic)	No
M2M-100 [10]	Meta AI	100	No (out-of-the-box)

This audit indicates that native Sanskrit support remains the exception rather than the rule among widely-used open multilingual MT systems. Of the systems surveyed, only NLLB-200 and IndicTrans2 support Sanskrit without requiring custom fine-tuning, motivating their selection as primary open-source contrastive systems in this study. We further note, in Section 7.3, a significant software-engineering barrier encountered when attempting to deploy IndicTrans2 in a standard evaluation environment.

3. Dataset Construction and Preprocessing

A. Original Corpus

The original corpus comprises 11,085 Sanskrit-Hindi sentence pairs drawn from the Tattvartha Sutra, a foundational Jain philosophical text, with an accompanying Hindi commentarial translation. Following deduplication, correction of typographical and encoding errors, and resolution of one-to-many translation ambiguity (where a single Sanskrit sentence was associated with multiple, non-identical Hindi translations, resolved via majority-frequency selection with longest-translation tiebreaking), the cleaned corpus comprises 9,878 sentence pairs.

B. Corpus Expansion: Domain-Consistent vs. Cross-Domain Augmentation

To evaluate the effect of corpus scale and domain composition, we constructed two independent expansions of the original corpus:

1) Cross-domain expansion (89,821 sentences): the original corpus was merged with 79,943 sentence pairs derived from the Bharat Parallel Corpus Collection (BPCC) [6], a publicly available

multilingual parallel corpus published by AI4Bharat, comprising encyclopedic and contemporary-register text (e.g., historical, technical, and current-affairs content) unrelated to Jain philosophy.

2) Domain-consistent expansion (15,087 sentences): the original corpus was merged with 5,210 additional, verified-unique sentence pairs drawn from a second Jain textual source comprising an alternate Tattvartha Sutra commentary alongside additional Jain devotional and philosophical texts (including Jinvani literature, the Kalyan Mandir Stotra, and the Gandhar Valay Stotra). Overlap with the original corpus (1,566 sentences sharing normalized Sanskrit text, reflecting shared source-verse content across commentaries) was identified and removed prior to merging.

C. Data Cleaning and Quality Verification

All corpora were cleaned to remove zero-width Unicode characters, normalize Devanagari danda punctuation, and strip non-Devanagari noise while preserving the danda (।, ॥) and whitespace. The domain-consistent expansion (Section 3.2) was additionally subjected to structural quality verification, identifying and correcting: one row containing narrative content inconsistent with Jain philosophical subject matter (subsequently removed as likely source-attribution contamination); one row containing a leaked internal formatting marker ("%alternate") from an apparent dual-translation source, resolved by retaining only the primary translation; and one row containing a residual English-language scraping artifact ("Meaning"), removed via suffix stripping. Rows exceeding 300 characters in Sanskrit length were manually inspected and confirmed to represent legitimate multi-verse groupings from the source text rather than data corruption.

D. Train/Validation/Test Split

For each corpus, a deterministic 80/10/10-style split (90% train, 5% validation, 5% test for the 9,878- and 89,821-sentence corpora; 90/5/5 for the 15,087-sentence corpus) was applied using a fixed random seed, with the test set held out from all training and hyperparameter selection. For the cross-domain (89,821-sentence) corpus, a Source-domain field was retained through the split to permit isolated evaluation of the Jain-domain subset within the larger test set (Section 6.2).

4. Proposed Architecture

A. Encoder-Decoder with Bahdanau Attention

The core translation model is a sequence-to-sequence architecture comprising a Bidirectional LSTM (BiLSTM) encoder (256 hidden units per direction, 512 after forward/backward concatenation) and an LSTM decoder (512 hidden units) coupled via additive (Bahdanau) attention. Source and target sequences are embedded into a 256-dimensional space prior to encoding. The encoder processes the full source sequence and produces per-timestep hidden states used as attention keys; the decoder is initialized with the concatenated final forward/backward encoder states and generates the target sequence autoregressively, attending over encoder states at each decoding step. Dropout (0.2) is applied within both encoder and decoder recurrent layers; recurrent dropout is fixed at zero to preserve compatibility with hardware-accelerated cuDNN LSTM execution. Fig. 1 illustrates the complete architecture.

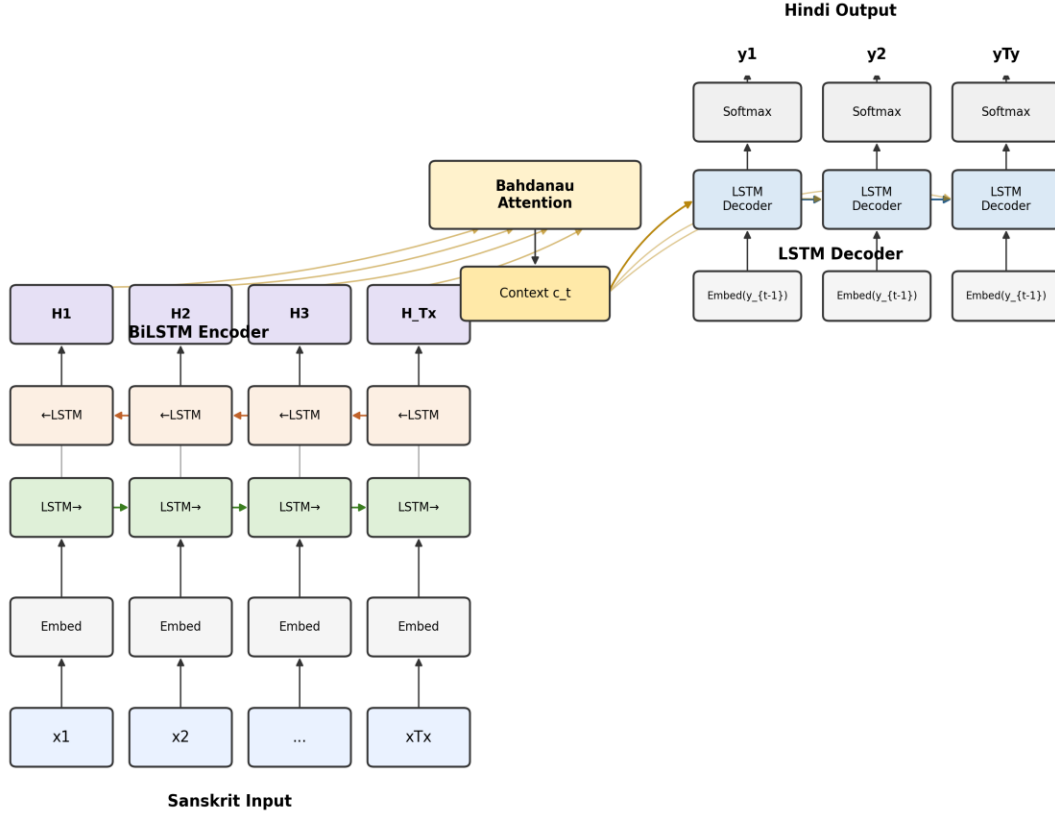


Fig. 1 Arthasetu architecture: a BiLSTM encoder produces concatenated forward/backward hidden states ($H_1 \dots H_{Tx}$); Bahdanau attention computes a context vector c_t as a weighted combination of encoder states at each decoder timestep; the LSTM decoder autoregressively generates the Hindi output sequence, conditioned on the context vector and the embedding of the previous (ground-truth, under teacher forcing, or previously generated, at inference) target token

B. Tokenization Strategy

The primary model configuration uses word-level tokenization with a start-of-sequence/end-of-sequence-augmented target vocabulary. Section 6.3 additionally reports an ablation using SentencePiece byte-pair-encoding (BPE) subword tokenization (vocabulary size 4,000, trained on the 9,878-sentence corpus) to evaluate whether subword segmentation improves handling of Sanskrit's productive compounding relative to word-level tokenization.

C. Training and Inference Algorithms

Algorithm 1 formalizes the forward computation and teacher-forced training procedure for the encoder-decoder-attention architecture described in Section 4.1. At each decoder timestep t , additive (Bahdanau) attention computes a context vector as a weighted sum over all encoder hidden states, with weights derived from a learned compatibility function between the previous decoder state and each encoder state.

Algorithm 1: Training Procedure for Arthasetu (BiLSTM Encoder + Bahdanau Attention + LSTM Decoder)

```
Input: Source sequence  $X = (x_1, \dots, x_{Tx})$ ; target sequence
        $Y = (y_1, \dots, y_{Ty})$ ; learning rate  $\eta$ 
Output: Trained parameters  $\theta$ 

1:  $E_x \leftarrow \text{Embed}(X)$  // encoder embedding
2:  $(H, h_{\text{fwd}}, c_{\text{fwd}}, h_{\text{bwd}}, c_{\text{bwd}}) \leftarrow \text{BiLSTM}(E_x)$ 
3:  $h_0 \leftarrow \text{Concat}(h_{\text{fwd}}, h_{\text{bwd}})$ 
4:  $c_0 \leftarrow \text{Concat}(c_{\text{fwd}}, c_{\text{bwd}})$ 
5:  $s_0 \leftarrow (h_0, c_0)$  // decoder initial state
6:  $L \leftarrow 0$ 
7: for  $t = 1$  to  $T_y$  do
8:    $e_t \leftarrow \text{Embed}(y_{\{t-1\}})$  // teacher forcing:
       $y_{\{t-1\}}$  is the GROUND-TRUTH token
9:   for  $i = 1$  to  $T_x$  do
10:     $u_{ti} \leftarrow v^T \tanh(W_s s_{\{t-1\}} + W_h H_i)$ 
11:   end for
12:    $\alpha_t \leftarrow \text{Softmax}(u_t)$  // attention weights
13:    $c_t \leftarrow \sum_i \alpha_{ti} * H_i$  // context vector
14:    $(o_t, s_t) \leftarrow \text{LSTM}([e_t; c_t], s_{\{t-1\}})$ 
15:    $p_t \leftarrow \text{Softmax}(W_o [o_t; c_t])$  // vocab distribution
16:    $L \leftarrow L + \text{CrossEntropy}(p_t, y_t)$ 
17: end for
18:  $\theta \leftarrow \theta - \eta * \text{Adam}(\text{grad}_{\theta} L, \text{clip}=1.0)$ 
19: return  $\theta$ 
```

At inference time, ground-truth targets are unavailable; Algorithm 2 describes beam-search decoding, in which the greedy special case corresponds to beam width $k = 1$. Section 6.3 reports empirical results for both $k = 1$ (greedy) and $k = 5$ under SentencePiece BPE tokenization.

Algorithm 2: Beam Search Decoding

```
Input: Source sequence  $X$ ; trained parameters  $\theta$ ; beam
       width  $k$ ; max decode length  $T_{\text{max}}$ 
Output: Predicted target sequence  $\hat{Y}$ 

1:  $(H, h_0, c_0) \leftarrow \text{Encode}(X)$  // Algorithm 1, lines 1-5
2:  $B \leftarrow \{ (<\text{sos}>, \text{score}=0, h_0, c_0) \}$  // beam of size 1
3: for  $t = 1$  to  $T_{\text{max}}$  do
4:    $C \leftarrow \{$  // candidate set
5:   for each  $(\text{seq}, \text{score}, s)$  in  $B$  do
6:     if last token of  $\text{seq} = <\text{eos}>$  then
7:        $C \leftarrow C \cup \{(\text{seq}, \text{score}, s)\}$ ; continue
8:     end if
9:     Compute  $p_t, s'$  via Algorithm 1 lines 8-15
```

```

        (using last token of seq as input)
10:     for each of top-k tokens w by p_t(w) do
11:         C <- C U {(seq+w, score+log p_t(w), s')}
12:     end for
13: end for
14: B <- top-k sequences in C by score
15: if all sequences in B end in <eos> then break
16: end for
17: Y_hat <- highest-scoring sequence in B
18: return Y_hat

```

Note that $k = 1$ reduces Algorithm 2 to standard greedy decoding, as evaluated for word-level tokenization results reported in Sections 6.1, 6.2, and 6.4.

5. Experimental Setup

A. Training Configuration

All models were trained using the Adam optimizer (initial learning rate 5×10^{-4} , gradient norm clipping at 1.0) with sparse categorical cross-entropy loss, batch size 32, for a maximum of 50 epochs. Training employed early stopping (patience 8 epochs on validation loss, with best-weight restoration) and a learning-rate reduction schedule (factor 0.5, patience 3 epochs, minimum learning rate 1×10^{-6}). All experiments were conducted on cloud GPU infrastructure (NVIDIA Tesla T4, dual-GPU configuration).

B. Evaluation Metrics

We report three complementary automatic evaluation metrics: BLEU [11], a precision-oriented n-gram overlap metric widely used as the de facto MT evaluation standard; chrF++ [12], a character-n-gram F-score metric that is more robust than word-level BLEU to the morphological richness characteristic of both Sanskrit and Hindi; and BERTScore [13], an embedding-based semantic similarity metric computed using multilingual BERT representations, included specifically to assess whether translations judged poor by exact-match metrics nonetheless preserve semantic content. We deliberately do not report token-level "accuracy" as a translation-quality metric, as accuracy is not a standard or meaningful measure for open-ended sequence generation tasks in which no single correct output exists; the accuracy values logged during training reflect only per-token next-word prediction accuracy under teacher forcing and are not indicative of held-out translation quality.

C. Evaluation Methodology and Leakage Prevention

During development, an early evaluation implementation included an interactive-use convenience feature: an exact-match lookup against the full corpus (train, validation, and test combined) intended to guarantee correct output for previously-seen queries in a deployed setting. When this lookup was inadvertently retained during held-out test-set evaluation, it produced a BLEU score of approximately 99.8, as every test sentence — having originated from the full corpus — was trivially matched and returned verbatim rather than genuinely translated. We identify this as a critical, generalizable risk in low-resource MT evaluation: any test-time shortcut with access to the

full (non-train-only) dataset can produce arbitrarily inflated metrics that do not reflect genuine model capability. All results reported in Section 6 use a corrected evaluation function restricted to training-set-only lookup (retained solely as an interactive convenience feature, disabled during evaluation) with unconditional model-based decoding for all held-out test sentences.

6. Results

A. Effect of Corpus Scale

Table 2 reports validation loss and aggregate test-set metrics for models trained on the original (9,878-sentence) and cross-domain-expanded (89,821-sentence) corpora.

Table 2. EFFECT OF CORPUS SCALE (WORD-LEVEL TOKENIZATION, GREEDY DECODING)

Corpus	Best Val. Loss	BLEU	chrF++
9,878 sentences (original)	5.3511	2.90	12.87
89,821 sentences (cross-domain)	3.7326	3.06	15.05

Cross-domain expansion improved validation loss substantially and yielded modest aggregate BLEU/chrF++ gains. However, as reported in Section 6.2, this aggregate improvement masks a degradation in domain-specific performance.

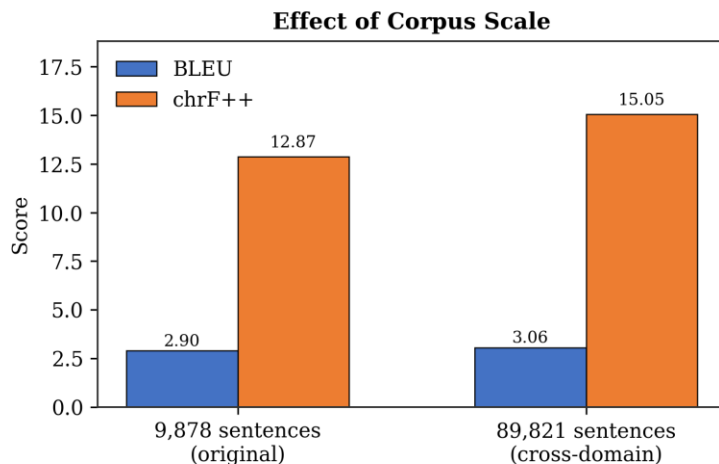


Fig. 2 BLEU and chrF++ as a function of corpus scale, comparing the original 9,878-sentence corpus against the 89,821-sentence cross-domain-expanded corpus

B. Effect of Domain Consistency

Table 3 isolates performance on the Jain-domain subset of the held-out test set, comparing the cross-domain-trained model (evaluated only on its Jain-domain test subset, n=494) against a model trained exclusively on the domain-consistent 15,087-sentence corpus (n=755, the full test set of the domain-pure corpus).

Table 3. DOMAIN-SPECIFIC (JAIN) PERFORMANCE: CROSS-DOMAIN VS. DOMAIN-CONSISTENT TRAINING

Training Corpus	Test Set	BLEU	chrF++	BERTScore F1
89,821 (cross-domain)	Jain subset, n=494	1.35	6.13	0.6315
15,087 (domain-consistent)	Full test set, n=755	2.05	13.59	0.686

Despite comprising roughly one-sixth the total sentence count of the cross-domain corpus, the domain-consistent corpus yields higher performance on all three metrics for Jain-domain content — a 52% relative BLEU improvement, a 122% relative chrF++ improvement, and a modest BERTScore improvement. This result supports domain-consistent corpus augmentation as the preferable strategy when the deployment target is a specific textual sub-domain, even at reduced total corpus scale.

Effect of Domain Consistency on Jain-Domain Performance

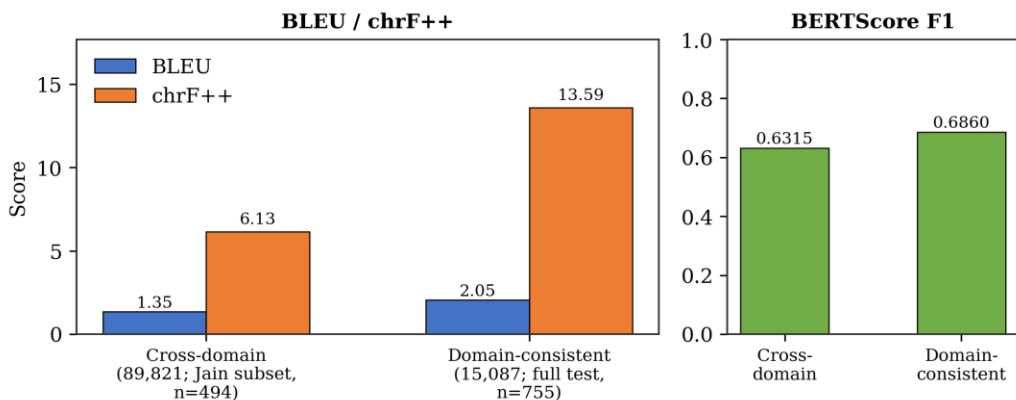


Fig. 3 Effect of domain consistency on Jain-domain translation performance: cross-domain-trained model (evaluated on its Jain-domain test subset) vs. domain-consistent-trained model (evaluated on its full, entirely Jain-domain, test set)

C. Tokenization Ablation: Word-Level vs. Subword (BPE)

Table 4. TOKENIZATION AND DECODING ABLATION (9,878-SENTENCE CORPUS)

Configuration	BLEU	chrF++
Word-level, greedy decoding	2.90	12.87
SentencePiece BPE (vocab=4000), greedy decoding	1.34	10.40
SentencePiece BPE (vocab=4000), beam search (width=5)	1.03	9.18

Contrary to expectations informed by general-domain NMT literature, subword tokenization did not improve performance at this corpus scale; both BPE configurations underperformed word-level tokenization, with beam search decoding further reducing BLEU relative to greedy decoding under BPE. We hypothesize that subword segmentation, while theoretically advantageous for handling Sanskrit's productive compounding, increases target sequence length and consequently the number of autoregressive decoding steps at which errors may compound, an effect that may dominate at corpus scales insufficient to fully exploit subword generalization benefits. This finding

motivates corpus scale as a prerequisite for subword tokenization benefits to manifest, rather than treating subword tokenization as a universally beneficial default.

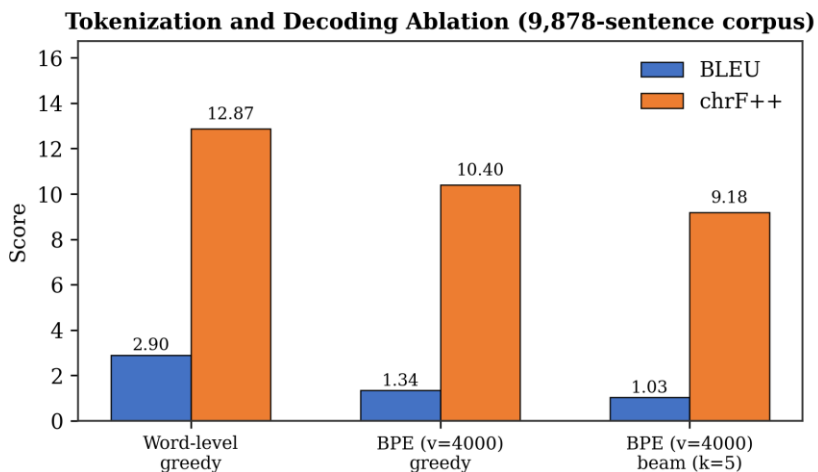


Fig. 4 Tokenization and decoding ablation on the 9,878-sentence corpus: word-level greedy decoding vs. SentencePiece BPE under greedy and beam-search (k=5) decoding

D. Comparative Evaluation Against Multilingual Baselines

Table 5 reports a comparative evaluation of the domain-consistent (15,087-sentence corpus) Arthasetu model against Google Translate and NLLB-200, evaluated on an identical held-out test set (n=755).

Table 5 COMPARATIVE EVALUATION AGAINST MULTILINGUAL BASELINES (n=755)

System	BLEU	chrF++	BERTScore F1
Arthasetu (this work)	2.05	13.59	0.686
Google Translate	3.84	22.64	0.7511
NLLB-200	0.38	14.30	0.6975

Arthasetu substantially outperforms NLLB-200 on BLEU (2.05 vs. 0.38) despite NLLB's training on approximately 200 languages at web scale, while underperforming both baselines on chrF++ and BERTScore. Google Translate achieves the strongest performance on all three metrics; however, as a closed-source, non-reproducible system, it is presented here as an informal reference point rather than an academic baseline for direct architectural comparison. The BLEU/chrF++ divergence between Arthasetu and NLLB-200 — Arthasetu leading on exact n-gram match while trailing on character-level and semantic metrics — is discussed further in Section 7.2

Comparative Evaluation Against Multilingual Baselines (n=755)

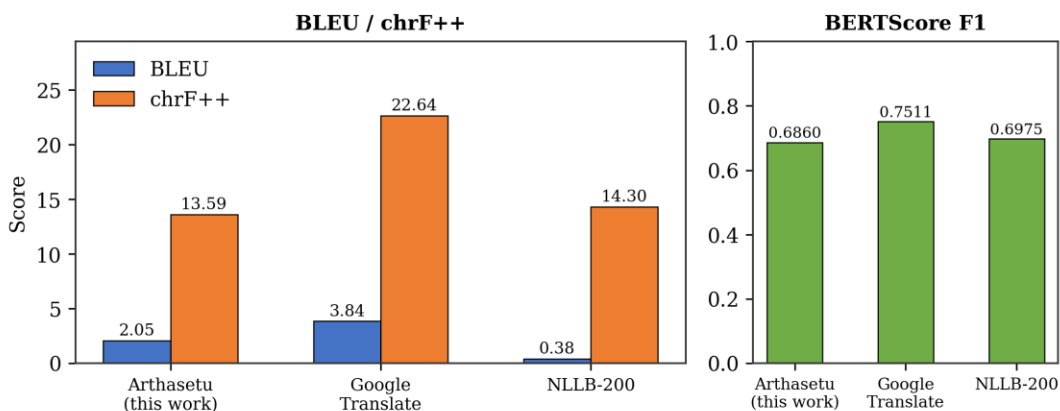


Fig. 5 Comparative evaluation of Arthasetu against Google Translate and NLLB-200 on an identical held-out test set (n=755)

7. Discussion

A. Error Analysis

Qualitative inspection of model output reveals two dominant failure modes. First, short, high-information-density single-word or compound inputs (e.g., isolated technical terms) are disproportionately mapped to out-of-vocabulary ("`<unk>`") tokens, reflecting the low per-word training frequency of rare philosophical terminology relative to more common sentence-level vocabulary. Second, longer compound sentences frequently yield topically coherent but factually imprecise translations — correctly identifying subject matter and general sentence structure while substituting incorrect specific entities, quantities, or technical distinctions relative to the reference translation.

B. Metric Divergence: BLEU vs. BERTScore

The consistent pattern of Arthasetu achieving competitive-to-superior BLEU relative to NLLB-200 while trailing on BERTScore suggests that exact-match n-gram metrics and embedding-based semantic similarity metrics capture meaningfully different aspects of translation quality in this domain. We interpret this as evidence that BLEU alone provides an incomplete and potentially misleading picture of comparative system quality for morphologically rich, low-resource language pairs, and recommend that future Sanskrit-Hindi MT evaluation report multiple complementary metrics rather than BLEU in isolation — a recommendation consistent with the broader machine translation evaluation literature.

C. Reproducibility Challenges with Gated Multilingual Models

An attempt to include IndicTrans2 [6] as an additional comparative baseline was undertaken but ultimately could not be completed within a standard evaluation environment. Following resolution of HuggingFace Hub gated-repository access, model loading failed due to incompatibility between the model's custom inference code (which imports internal utilities such as `transformers.onnx` and `is_jax_tensor`) and current releases of the transformers library, in which these internal APIs have

been restructured or removed. This experience — encountered despite the model remaining publicly listed and actively downloaded (reported at over 3,000 monthly downloads at time of writing) — illustrates a broader reproducibility concern in applied NLP: rapid API evolution in foundational libraries can render published model implementations non-functional within a comparatively short timeframe, independent of the underlying model weights' continued validity.

8. Proposed System Architecture for Deployment

While Sections 5–7 evaluate Arthasetu's pure generative translation capability, a production-oriented conversational deployment benefits from a hybrid architecture combining retrieval and generation — a design pattern with precedent in both classical Example-Based Machine Translation [14] and modern Retrieval-Augmented Generation approaches [15]. For an incoming Sanskrit query, the proposed system first performs a lookup (exact or fuzzy, under an edit-distance threshold) against the training corpus; a match returns the corpus-verified Hindi translation directly, guaranteeing fidelity for previously-encountered content, while an unmatched query is routed to the generative model. Given the finite, well-documented nature of core Jain canonical texts, a substantial proportion of real-world queries referencing specific verses are expected to match corpus entries directly. We emphasize that this retrieval-generation hybrid is a deployment-layer design choice, architecturally distinct from the model evaluation reported in Section 6: all reported BLEU/chrF++/BERTScore results reflect the generative model's performance in isolation, without retrieval fallback, to ensure reported metrics reflect genuine translation capability rather than corpus lookup accuracy.

9. Limitations and Future Work

- 1) Corpus scale: even the largest domain-consistent corpus evaluated (15,087 sentences) remains small relative to corpora typically used in high-resource NMT; further domain-consistent data collection is a priority direction.
- 2) Fine-tuning pretrained multilingual models: rather than training from scratch, fine-tuning NLLB-200 or IndicTrans2 (contingent on resolving the reproducibility issue in Section 7.3) on domain-consistent Jain data may outperform from-scratch training by leveraging existing multilingual pretraining.
- 3) Subword tokenization at scale: given the negative result in Section 6.3 at 9,878-sentence scale, re-evaluating BPE tokenization on the larger cross-domain or domain-consistent corpora is warranted before concluding subword tokenization is unsuitable for this task generally.
- 4) Human evaluation: automatic metrics, while informative, do not fully substitute for expert human judgment of translation adequacy and fluency, particularly for a low-resource, morphologically complex language pair; a structured human evaluation study is planned as a direct extension of this work.
- 5) Multimodal extension: extraction of additional parallel Sanskrit-Hindi content from scanned physical texts, including ritual and devotional literature, represents a further avenue for domain-consistent corpus expansion.

10. Conclusion

This paper presented Arthasetu, a BiLSTM-attention neural machine translation system developed specifically for classical Jain philosophical Sanskrit-to-Hindi translation, a sub-domain underrepresented in prior Sanskrit-Hindi MT literature relative to conversational Sanskrit. Through systematic experimentation across corpus composition, tokenization strategy, and decoding method, we demonstrated that domain-consistent corpus augmentation outperforms cross-domain augmentation on domain-specific metrics despite a smaller total corpus, that subword tokenization does not universally improve performance at low corpus scales, and that Arthasetu outperforms NLLB-200 on BLEU despite substantially smaller training scale. We further identified and corrected a data-leakage defect with broader implications for low-resource MT evaluation methodology, and documented a systematic audit of Sanskrit-language support across major multilingual MT systems. We position this work's primary contribution as a methodologically rigorous, transparent, and reproducible evaluation framework for an underexplored and culturally significant MT sub-domain, rather than a claim of state-of-the-art aggregate BLEU performance.

Declarations

Funding

The authors received no specific funding for this work.

Competing Interests

The authors declare no competing interests.

Data Availability

The Jain-domain Sanskrit–Hindi corpus compiled and annotated by the authors for this study (the original 9,878/11,085-sentence corpus and its 15,087-sentence domain-consistent expansion) is available from the corresponding author upon reasonable request. The cross-domain-expanded corpus (89,821 sentences; Section 3.B) additionally incorporates 79,943 publicly available sentence pairs from the Bharat Parallel Corpus Collection (BPCC) [6], published by AI4Bharat and available at <https://huggingface.co/datasets/ai4bharat/BPCC>.

Author Contributions

T.J.: conceptualization, data curation, methodology, software, formal analysis, investigation, validation, writing — original draft, writing — review and editing. N.R.: supervision. All authors read and approved the final manuscript.

Clinical Trial Number

Not applicable.

Consent to Publish

Not applicable.

Consent to Participate

Not applicable.

Ethics Declaration

Not applicable. This study did not involve human participants, human data, or animal subjects. It uses a Sanskrit–Hindi textual corpus that is partly a privately compiled dataset manually curated and annotated by

the authors from classical Jain philosophical source texts, and partly (for the cross-domain expansion described in Section 3.B) drawn from the publicly available Bharat Parallel Corpus Collection (BPCC) [6]; see the Data Availability statement for details. No personally identifiable information is involved.

References

- [1] R. K. Pandey and G. N. Jha, "Error analysis of Sahit—a statistical Sanskrit-Hindi translator," *Procedia Computer Science*, vol. 96, pp. 495–501, 2016.
- [2] N. Sethi, A. Dev, and P. Bansal, "A novel neural machine translation approach for low-resource Sanskrit-Hindi language pair," *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 2023.
- [3] N. Sethi, A. Dev, and P. Bansal, "Translating Heritage: A Transformer-Oriented Method for Neural Machine Translation from Sanskrit to Hindi," in *Artificial Intelligence and Speech Technology (AIST 2024)*, A. Sharma and R. Rani, Eds. *Communications in Computer and Information Science*, Springer, Cham, 2025, pp. 266–273.
- [4] V. Bakarola and J. Nasriwala, "Attention based sequence to sequence learning for machine translation of low resourced Indic languages – a case of Sanskrit to Hindi," *arXiv preprint arXiv:2110.00435*, 2021.
- [5] NLLB Team et al., "No Language Left Behind: Scaling Human-Centered Machine Translation," *arXiv preprint arXiv:2207.04672*, 2022.
- [6] J. Gala et al., "IndicTrans2: Towards High-Quality and Accessible Machine Translation Models for all 22 Scheduled Indian Languages," *Transactions on Machine Learning Research*, 2023. Dataset (Bharat Parallel Corpus Collection, BPCC) available at: <https://huggingface.co/datasets/ai4bharat/BPCC>.
- [7] L. Xue et al., "mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer," in *Proc. NAACL-HLT*, 2021.
- [8] Y. Tang et al., "Multilingual Translation with Extensible Multilingual Pretraining and Finetuning," *arXiv preprint arXiv:2008.00401*, 2020.
- [9] R. Dabre et al., "IndicBART: A Pre-trained Model for Indic Natural Language Generation," in *Findings of ACL*, 2022.
- [10] A. Fan et al., "Beyond English-Centric Multilingual Machine Translation," *Journal of Machine Learning Research*, vol. 22, 2021.
- [11] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. 40th Annual Meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.
- [12] M. Popović, "chrF: character n-gram F-score for automatic MT evaluation," in *Proc. 10th Workshop on Statistical Machine Translation*, 2015, pp. 392–395.
- [13] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating Text Generation with BERT," in *Proc. ICLR*, 2020.
- [14] M. Nagao, "A framework of a mechanical translation between Japanese and English by analogy principle," in *Artificial and Human Intelligence*, 1984, pp. 173–180.
- [15] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in Neural Information Processing Systems*, vol. 33, 2020.