Naamah: A Large Scale Synthetic Sanskrit NER Corpus via DBpedia Seeding and LLM Generation

Annarao Kulkarni and Akhil Rajeev P
Centre for Development of Advanced Computing (C-DAC), Bangalore
{akhil.rajeev, annarao}@cdac.in

Abstract

The digitization of classical Sanskrit literature is impeded by a scarcity of annotated resources, particularly for Named Entity Recognition (NER). While recent methodologies utilize generic Large Language Models (LLMs) for data augmentation, these approaches remain prone to error and often lack the reasoning depth required for classical grammar. In this work, we introduce **Naamah**, a high quality silver standard Sanskrit NER dataset comprising **102,942** sentences. We propose a methodology that combines entity extraction from **DBpedia** with the generative capabilities of a **24B parameter hybrid reasoning model** to create grammatically natural and synthetically diverse training data. We utilize this dataset to benchmark two transformer architectures: the massive multilingual **XLM RoBERTa** and the parameter efficient **IndicBERTv2**. Our experiments reveal a key insight: while both models scale well with synthetic data, IndicBERTv2 qualitatively outperforms XLM RoBERTa in entity identification and classification. On a fixed split of 92,647 train and 10,295 validation examples, IndicBERTv2 achieves the best validation F1 of 0.9615, outperforming XLM R's 0.9506 while remaining substantially lighter for deployment. We demonstrate that the generic tokenizer of XLM R fractures Sanskrit terms, whereas the domain adapted tokenizer of IndicBERTv2 preserves semantic integrity.

Keywords: Sanskrit NER, Synthetic Data, Language Models, Low Resource NLP, Dataset Creation

The **Naamah** dataset is publicly available at https://huggingface.co/datasets/akhil2808/Naamah.

1. Introduction

Sanskrit is central to South Asian intellectual history, yet modern NLP resources for Sanskrit remain sparse relative to contemporary high resource languages. Extracting structured information from this corpus constitutes a significant challenge in Digital Humanities. Named Entity Recognition (NER) serves as the foundational step for downstream tasks such as Knowledge Graph construction, relation extraction, digital philology, and historical prosopography. However, developing NER systems for Sanskrit is complicated by two primary factors: intrinsic linguistic complexity and a scarcity of annotated resources.

Sanskrit is a Morphologically Rich Language (MRL) characterized by extensive agglutination and inflection. Unlike English, where word order largely determines syntax, Sanskrit relies on a complex system of case markers (*Vibhakti*). A single Named Entity (NE), such as *Rama*, can manifest in over 24 surface forms depending on its syntactic role. Furthermore, entities are often merged phonetically with adjacent words via *Sandhi*, obscuring their boundaries. Standard string matching techniques or rigid rule based systems often fail to capture this variance in unseen contexts.

The resource bottleneck is significant. Manual annotation requires high level domain expertise, making the creation of gold standard corpora slow and expensive. Existing datasets are often small, domain specific, or suffer from severe class imbalance. Current approaches to overcoming this include Cross Lingual Transfer projecting labels from high resource languages like English. However, projection methods introduce significant alignment noise due to structural mismatches. Similarly, utilizing generic LLMs for generation often yields errors because they lack domain specific grounding for Indic scripts.

In this paper, we leverage a hybrid reasoning model optimized for Indic languages to bridge this data gap. Our contributions are three fold:

1. **DBpedia Mining Strategy:** We detail a methodology for extracting diverse entity seeds from **DBpedia** using structured queries, ensuring broad coverage of Persons, Locations, and Organizations.

2. **The Naamah Corpus:** We introduce a silver standard dataset of **102,942** Sanskrit sentences generated via this model and refined through heuristic preprocessing. This method bypasses rigid grammar templates, allowing varied syntactic structures to emerge naturally.

3. **Benchmarking Insights:** We provide a comparative analysis of **XLM RoBERTa (Base)** vs. **IndicBERTv2** on a fixed split (92,647 and 10,295). We demonstrate that for classical languages, domain aligned tokenization is more critical than raw model scale.

Proceedings of WILDRE-8, pages 88–92
12 May 2026.

2. Related Work

2.1. Challenges in Cross Lingual Projection

A common approach to low resource NER involves projecting annotations from a source language to the target language via parallel corpora. For instance, the *Naamapadam* dataset (Mhaske et al., 2023) utilizes the Samanantar corpus to generate NER data for 11 Indic languages. Linguistic mismatches such as the divergence in word order and the lack of direct equivalents for Sanskrit case markers lead to alignment errors. Parallel alignment errors can propagate directly into label quality, especially with inflected forms. Our work circumvents this by generating data directly in the target language structure.

2.2. Sanskrit Computational Linguistics and NER

Rule-based paradigms have dominated traditional Sanskrit processing. Tools like the *Sanskrit Heritage Reader* (Goyal and Huet, 2016) excel at morphological analysis and segmentation. However, these tools lack the probabilistic flexibility required to disambiguate Named Entities in complex contexts where ambiguity is resolved through broader sentence semantics. While deep learning has been applied to segmentation (Hellwig and Nehrdich, 2020), contextual NER remains under-explored. Early efforts in Sanskrit NER largely relied on rule-based heuristics and dictionary lookups, which naturally struggle with out-of-vocabulary terms and extensive *Sandhi* (Murthy et al., 2008). Subsequent attempts have explored statistical models like Conditional Random Fields (CRFs) on limited, domain-specific corpora (Bhargava and Sharma, 2016). This research gap is further widened by the lack of inclusion in foundational datasets; for instance, the *Namapadam* dataset, which serves as the primary large-scale repository for NER in Indic languages, does not currently include Sanskrit.

2.3. Sanskrit Digital Resources

The development of robust NLP models for classical languages heavily depends on the availability of digitized texts and lexical frameworks. Several notable efforts have laid the groundwork for Sanskrit digital humanities. The *Digital Corpus of Sanskrit* (DCS) (Hellwig, 2010) provides an extensively annotated corpus for morphological and lexical analysis, while the *Göttingen Register of Electronic Texts in Indian Languages* (GRETIL) serves as a comprehensive repository of machine-readable foundational texts. Additionally, lexical resources like the *IndoWordNet* (Bhattacharyya, 2010) offer valuable semantic linkages. While these digital resources are invaluable for philological research, grammar formulation, and basic NLP tasks, they generally lack the dense, large-scale semantic annotations required for training modern deep-learning-based NER systems. This scarcity directly underscores the necessity for the synthetic data generation pipeline proposed in this work.

2.4. Synthetic Data Generation

Data augmentation is a standard technique in low resource NLP (Ding et al., 2020). The current trend relies heavily on generic LLMs (Wang et al., 2023), which can generate incorrect grammatical structures in low resource languages. Our work utilizes an LLM optimized specifically for Indic scripts, offering a domain grounded generative alternative.

3. Automated Entity Extraction from DBpedia

A critical challenge in synthetic data generation is ensuring the diversity of the lexicon. If the model observes only a handful of traditional names during training, it will simply memorize those tokens rather than learning the morphological context of a Named Entity. To address this, we propose leveraging **DBpedia**, a large scale multilingual knowledge base.

3.1. Knowledge Base Structure

DBpedia organizes knowledge as a graph of triples using the Resource Description Framework (RDF). This structure allows researchers to programmatically filter entities based on their ontology classes.

3.2. Extraction Methodology

By utilizing SPARQL, we extracted a broad spectrum of entities targeting three primary categories: Person, Location, and Organization.

To ensure morphological variety, the extraction included a diverse mix of both classical Indian entities and global entities (e.g., modern international locations, foreign political figures) transliterated into Devanagari script. Embedding transliterated names like *Giacomo Libera* or *Manfred Hake* alongside traditional Sanskrit entities prevents downstream NER models from relying on lexical familiarity, forcing them to learn the underlying syntactic patterns and case markers (*Vibhakti*) that designate an entity in a Sanskrit sentence.

4. The Naamah Corpus

Using the vetted entity lists, we developed our dataset. We shifted from a deterministic logic ap-

proach to an LLM driven generative pipeline to maximize syntactic fluidity.

4.1. Language Model Pipeline

Instead of relying on rigid, pre-programmed morphological engines that often struggle with the fluid nature of Sanskrit syntax, we utilised Sarvam M, a 24-billion-parameter hybrid reasoning model, heavily optimised for Indic languages.

Generation Process: The model was prompted to incorporate specific entity seeds from our DBpedia extraction into semantically coherent Sanskrit sentences. This generative approach allows for the natural emergence of appropriate case endings and phrasing that mimics authentic text better than brittle template only generation, yielding a wider variety of syntactic structures and inflectional realizations.

Preprocessing and Heuristics: To ensure the dataset could serve as a reliable silver standard, the raw output underwent a Python based preprocessing layer. Generated candidates are filtered using rule based checks for token label consistency, malformed output, and ambiguous boundaries. After filtering, we retain 102,942 high quality silver standard examples.

4.2. Dataset Characteristics and Statistics

The final dataset consists of **102,942** sentences structured in JSONL format, providing a substantial corpus for training and evaluation. It utilizes the standard BIO (Beginning-Inside-Outside) tagging scheme to represent entity boundaries. These tags are mapped to numeric identifiers via a `label2id` dictionary to facilitate model processing `"O": 0`, `"B-PER": 1`, `"I-PER": 2`, `"B-ORG": 3`, `"I-ORG": 4`, `"B-LOC": 5`, and `"I-LOC": 6`.

We performed a statistical analysis of the generated corpus (Table 1). The dataset relies on a highly diverse vocabulary, featuring **123,923 unique tokens** across a total volume of **732,267 tokens**. The average sentence length is 7.11 tokens.

Statistic	Value
Total Sentences	102,942
Train Split	92,647
Validation Split	10,295
Unique Tokens	123,923
Total Tokens	732,267
Average Sentence Length	7.11

Table 1: Core statistics of the Naamah corpus and split configuration used in experiments.

Entity Class	Count (B tags)
Person (PER)	90,452
Location (LOC)	22,290
Organization (ORG)	14,655
Total Entities	**127,397**

Table 2: Entity distribution in Naamah.

5. Experimental Setup

We benchmarked two state of the art transformer models on our dataset to evaluate their capacity to learn from synthetic Sanskrit data.

5.1. Models

1. **XLM RoBERTa (Base):** Serves as a strong multilingual baseline. It uses a large vocabulary (250k) and is often the default choice for low resource languages. However, its generic training data includes very little classical Sanskrit.

2. **IndicBERTv2 (MLM Only):** Provides an Indic focused compact alternative. It utilizes parameter sharing to reduce size to $\approx$ 130MB, making it suitable for edge deployment. Its vocabulary is optimized for Indic scripts.

5.2. Tokenization Strategy and De-Sandhi

Sanskrit is highly agglutinative, and authentic texts often feature virtually infinitely long string sequences due to complex *Sandhi* (phonetic fusions across word boundaries). While traditional Sanskrit NLP pipelines heavily rely on explicit de-sandhi preprocessing to separate these compound structures before tagging, our methodology evaluates the capacity of modern transformer tokenizers to handle raw, un-split text natively. Rather than applying a dedicated de-sandhi tool, we rely on the subword tokenization algorithms inherent to XLM-R and IndicBERTv2 to implicitly segment these agglutinated forms.

To handle the resulting fragmented sub-words during NER training, we employed a “Label First” alignment strategy. The BIO tag is assigned only to the first sub-token of an entity, and subsequent subtokens are masked with the ignore index (-100), a standard strategy for token classification with subword tokenizers. This forces the model to predict the entity type based on the root stem while implicitly learning the suffix structure and *Sandhi* fusions.

5.3. Training Configuration

Both models are fine tuned with Hugging Face Trainer. XLM R is trained for 3 epochs; IndicBERTv2 for 4 epochs. Batch size is 16. Learning rates are 2×10^{-5} (XLM R) and 3×10^{-5} (IndicBERTv2).

6. Results and Analysis

6.1. Quantitative Results

Both models achieved strong convergence on the synthetic test set (see Table 3). On a fixed validation split of 10,295 examples, IndicBERTv2 achieves the best validation F1 of 0.961451, outperforming XLM R's 0.950581.

Metric	XLM R	IndicBERTv2
Precision	0.949766	0.959563
Recall	0.951396	0.963345
F1 Score	0.950581	**0.961451**
Accuracy	0.985695	0.988897
Validation Loss	0.057814	0.054086

Table 3: Validation performance on Naamah (10,295 examples). IndicBERTv2 achieves the strongest overall NER quality.

6.2. Training Dynamics

XLM R converges in 3 epochs with gradual F1 gains (0.9366 $\rightarrow$ 0.9506). IndicBERTv2 converges in 4 epochs with stronger final validation F1 (0.9536 $\rightarrow$ 0.9615), indicating improved fit to Sanskrit entity morphology under the same data regime.

6.3. Qualitative Analysis: Tokenizer Model Fit

While aggregate scores demonstrate the viability of both models, qualitative error inspection shows a recurring issue for XLM R on inflected forms where suffix fragments receive unstable labels. We tested the models on sentences containing entities and structures not explicitly prevalent in the training data.

6.3.1. Failure Mode Analysis: Tokenizer Fragmentation

We observed a recurring failure mode in XLM R with complex agglutinated terms. For the input *"Kuruksetre"* (in Kurukshetra):

- **IndicBERTv2 Output:** Kuruksetre (Correctly classified as Location)
- **XLM R Output:** Kuruksetra (Location) + e (Incorrectly classified as Organization)

Analysis: XLM R's multilingual tokenizer, which is not optimized for Indic scripts, fractures the word into a root (*Kuruksetra*) and a suffix (*e*). The self attention mechanism treats the suffix *e* as a separate token. Without specific pre training on Sanskrit morphology, XLM R falsely predicts that this dangling suffix is an Organization. In contrast, IndicBERTv2 handles these sub word transitions coherently, recognizing that the suffix modifies the root and maintaining the Location tag across the entire span. IndicBERTv2 is more consistent on complete entity spans, supporting the hypothesis that Indic oriented tokenization better preserves morphological cues crucial for Sanskrit NER.

7. Discussion

A significant advantage of the proposed approach is that it effectively bypasses the requirement of manual annotation for Named Entities. Naamah demonstrates a practical path to scale labeled data for classical languages where expert annotation is scarce. While silver standard data does not replace gold corpora, it provides a strong foundation for pretraining and supervised transfer.

The results suggest that tokenizer language alignment is a primary factor for Sanskrit NER, often more influential than parameter count alone. For practitioners, this implies that compact domain adapted models can outperform larger multilingual encoders when script and morphology differ substantially from pretraining distributions.

8. Limitations and Future Work

Naamah is synthetic and inherits biases from source entities, prompting templates, and filtering heuristics. As a preliminary study, this work opens several avenues for critical improvement to transition from a silver standard to a production grade system:

8.1. Complex Sandhi Resolution

Future work will include targeted stress testing for complex Sandhi. While the generative LLM approach captures basic morphological fusions naturally, authentic Sanskrit literature is dominated by highly complex *Sandhi* (phonetic fusions across multiple words).

8.2. Gold Standard Evaluation

Future work will focus on evaluating manually annotated texts through hybrid training with an expert-validated gold subset. We plan to benchmark the

pre-trained **IndicBERTv2** model against excerpts from classical and contemporary sanskrit texts. This addresses the critical lack of Sanskrit support in modern NER datasets like *NamaPaadam* and the conceptual absence of "Organization" entities in classical corpora by adapting the annotation schema for historical contexts.

9. Conclusion

In this work, we introduced **Naamah**, a large-scale, silver-standard Sanskrit Named Entity Recognition dataset comprising 102,942 synthetically generated sentences. To overcome the critical scarcity of annotated classical Sanskrit texts, we developed a novel data generation pipeline that combined structured entity mining from DBpedia with the generative capabilities of a 24-billion-parameter hybrid reasoning LLM optimized for Indic languages. This methodology allowed us to bypass rigid, rule-based grammatical templates, resulting in a morphologically diverse and syntactically natural corpus.

We subsequently utilized this dataset to benchmark two distinct transformer architectures on a fixed split of 92,647 training and 10,295 validation examples. Our evaluations demonstrated that the parameter-efficient IndicBERTv2 achieved the highest validation F1 score (0.9615), outperforming the much larger, multilingual XLM-RoBERTa (0.9506). Crucially, our qualitative analysis revealed that generic multilingual tokenizers frequently fracture complex, agglutinated Sanskrit terms, leading to misclassification. In contrast, domain-adapted tokenization successfully preserves entity boundaries and semantic integrity. Ultimately, Naamah provides a robust foundational resource for advancing Sanskrit computational linguistics, demonstrating that language-aligned tokenization and targeted synthetic generation can effectively bridge the data gap for low-resource classical languages.

10. Bibliographical References

R. Bhargava and P. Sharma. 2016. Named entity recognition for sanskrit using conditional random fields. In *Proceedings of the International Conference on Natural Language Processing (ICON)*.

Pushpak Bhattacharyya. 2010. Indowordnet. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Bosheng Ding, Bill Yuchen Lin, Zhou Zhou, Zhefeng Chen, Bown Ren, and Yikang Zheng. 2020. Daga: Data augmentation with a generation approach for low-resource tagging tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6045–6057. Association for Computational Linguistics.

Pawan Goyal and Gérard Huet. 2016. Design and analysis of a sanskrit sandhi splitter. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1382–1392. The COLING 2016 Organizing Committee.

Oliver Hellwig. 2010. Dcs - the digital corpus of sanskrit. In *Linguistics, Archaeology and the Human Past, Occasional Paper 9*, Kyoto, Japan. Research Institute for Humanity and Nature.

Oliver Hellwig and Sebastian Nehrdich. 2020. Sanskrit segmentation with lstm networks. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5324–5332. European Language Resources Association.

Arnav Mhaske, Harshit Kedia, Sumanth Doddapaneni, and Mitesh M Khapra. 2023. Naamapadam: A large-scale named entity recognition dataset for indian languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5393–5414. Association for Computational Linguistics.

H. A. Murthy et al. 2008. Rule-based named entity recognition for indian languages. In *Proceedings of the IJCNLP Workshop on NER for South and South East Asian Languages*.

Shuhe Wang, Xiaofei Sun, and Jiwei Li. 2023. Gptner: Named entity recognition via large language models. *arXiv preprint arXiv:2304.10428*.