SAMVARDHINI
PEER REVIEWED MULTILINGUAL E-RESEARCH JOURNAL
(ISSN ONLINE: - 2583-7176)
https://samvardhini.in
https://www.samskritpromotion.in

SAMVARDHINI :- 01/06/2026
Volume – 8
Issue – 1
(ISSN ONLINE:- 2583-7176)
https://samvardhini.in

Artificial Intelligence, Sanskrit and the Road Ahead

Suraj Yadav

Preservation, Pedagogy, and the Long Case for Programming in Sanskrit

कृत्रिमबुद्धिः संस्कृतं च — भविष्यत्पथः

A Field Note for Researchers, Educators, and Curious Readers

July 2026

AT A GLANCE

Sanskrit's grammar is unusually rule-bound, which has made it a recurring point of interest for computer scientists since at least the 1980s. This piece walks through where AI is already helping Sanskrit scholarship, what a more mature version of that work might look like a decade from now, and the smaller, more speculative story of Sanskrit as an inspiration for programming languages themselves.

Suraj Yadav
AI Product Engineer
JetHat Cyber Security Pvt. Ltd.
Email :- ersuraj097@gmail.com

1. Why This Pairing Isn't as Odd as It Sounds

परिचयः — एतत् संयोजनं न आश्चर्यकरम्

Put 'ancient language' and 'artificial intelligence' in the same sentence and it can sound like a stretch — a nice metaphor rather than a real research programme. But the connection between Sanskrit and computational thinking has a genuine history, and it didn't start with a marketing pitch. It started with a NASA computer scientist named Rick Briggs, who published a paper in 1985 arguing that Sanskrit's grammatical structure had features worth studying for knowledge representation in AI systems. That paper still gets cited, still gets argued over, and still shapes how some researchers frame this whole field.

What follows isn't a survey of everything happening in this space — that would take a book, and a fast-moving one. It's a more grounded look at three things: where AI is genuinely useful for Sanskrit right now, what a realistic near future looks like, and the more niche but genuinely interesting question of what it would mean to write programs *in* Sanskrit, or to borrow its grammatical logic for language design.

2. Where AI Is Already Doing Real Work

वर्तमानस्थितिः

2.1 — Manuscript Digitisation

India's manuscript holdings are enormous — estimates commonly cited by the National Mission for Manuscripts put the number in the tens of millions, spread across temples, monasteries, universities, and private family collections. Most of that material has never been photographed, let alone transcribed. Optical character recognition trained specifically on Devanagari, Grantha, and Sharada scripts is slowly closing that gap, but it's slower going than OCR for printed Latin-script text, because manuscript handwriting varies enormously by scribe, period, and region. This is unglamorous, patient work, and it matters more than almost anything else on this list.

2.2 — Sandhi and Compound Splitting

Sanskrit fuses sounds across word boundaries (sandhi) and builds long compound words (samasa) that can carry the meaning of an entire English sentence in a single token. A human reader learns to unpack these through years of training; a parser has to be taught the same skill from data. Groups at institutions such as IIT Kanpur and Jawaharlal Nehru University have built rule-based and statistical splitters that get a meaningful share of this right, though edge cases — genuinely ambiguous compounds where even human scholars disagree — remain a real limitation, not a solved problem.

2.3 — Translation and Learning Tools

Machine translation from Sanskrit into Hindi, English, and other Indian languages has improved substantially over the last few years, largely by piggybacking on multilingual transformer models rather than Sanskrit-specific engineering. The quality is uneven — good enough for a student trying to get the gist of a passage, not yet reliable enough to replace a trained translator on a philosophically dense text. Language-learning apps and grammar-correction tools built on the same underlying models are a lower-

stakes, higher-payoff application: getting pronunciation and case-ending feedback wrong in a practice exercise is a minor inconvenience, not a scholarly error propagating into the literature.

3. A Realistic Look Ten Years Out

भविष्यदृष्टिः

It's tempting to describe the future of this field in sweeping terms — real-time conversational Sanskrit, AI systems that compose metrically correct verse, a searchable index of the entire surviving corpus. All of that is technically plausible over a ten-year horizon, and versions of each already exist in early form. But the more useful way to think about the next decade is in terms of what has to happen first.

- A shared, openly licensed dataset of annotated Sanskrit text — grammar-tagged, sandhi-split, cross-referenced — that multiple research groups can build on instead of each starting from scratch.
- Dedicated centres where Sanskrit scholars and machine-learning engineers sit in the same room, rather than exchanging papers across a disciplinary gap.
- Open-source parsers and OCR engines that smaller institutions without large compute budgets can actually run and improve.
- Sanskrit and IKS content integrated into mainstream data-science and computational-linguistics curricula, so the next generation of engineers isn't discovering Panini for the first time in a research paper.

None of this is a given. Funding for long-horizon, low-commercial-return language work tends to lose out to whatever has an obvious product attached to it. If this decade goes well, it's because institutions treated the dataset and infrastructure problem as seriously as the more exciting demo-able applications — the conversational assistants and verse generators that get the headlines but depend entirely on the unglamorous groundwork underneath them.

> *A single scholar's lifetime of manuscript study, complemented by tools that can scan ten thousand documents in the time it takes to read one — that is the honest promise here, not a replacement for the scholar.*
>
> *— Working description used across several IKS research proposals, 2023–2025*

Challenge	What It Actually Looks Like
Data Scarcity	Annotated Sanskrit corpora remain small compared with the training sets available for English, Mandarin, or Hindi.
Sandhi & Compounding	Phonological fusion and long compound formation make word-boundary detection a genuine research problem, not a solved one.
Script Diversity	Manuscripts survive in Devanagari, Grantha, Sharada, Newari, and regional scripts, each demanding its own OCR training.

Challenge	What It Actually Looks Like
Institutional Silos	Sanskrit departments and computer-science departments rarely share a corridor, let alone a research grant.
Compute & Funding	Long-horizon, low-commercial-return language work competes poorly against funding aimed at mainstream languages.

4. The More Speculative Question: Programming in Sanskrit

संस्कृते प्रोग्रामिंग्-विषये चिन्तनम्

This is the part of the conversation that tends to get overstated, so it's worth being precise about what is and isn't true. Nobody is writing production software in Sanskrit today, and there is no mainstream Sanskrit-based programming language in serious industrial use. What does exist is a genuine and long-running line of research into why Sanskrit's grammar — specifically Panini's Ashtadhyayi, a set of roughly four thousand interlocking rules composed around the 4th century BCE — resembles a formal generative grammar in ways that predate, and arguably anticipated, ideas later formalised in computer science.

4.1 — What Briggs Actually Argued

Rick Briggs' 1985 paper, published in AI Magazine, proposed that Sanskrit's explicit case-marking system could serve as a semantic representation format for AI knowledge bases — a way of encoding 'who did what to whom' without the ambiguity that plagues natural-language parsing in English. It's a narrower and more careful claim than the popular version of the story ('Sanskrit is the ideal language for AI'), which has circulated online in a much less rigorous form. The academic claim is about representational clarity for a specific class of problems, not a wholesale endorsement of Sanskrit as a programming substrate.

4.2 — Panini as a Rewrite System

Where the analogy holds up best is in Panini's method itself. The Ashtadhyayi is organised as an ordered system of rules, meta-rules, and exception-handling that determines which rule fires when several could apply — a structure computer scientists have compared to context-free grammars and to the kind of rule-precedence logic found in modern parser generators. This is why computational linguists studying Sanskrit often end up building software that treats Panini's grammar as something closer to an executable specification than a descriptive textbook.

4.3 — What's Actually Being Built

Thread of Work	What It Involves
Śābdika / rule-engines	Panini's ~4,000 sutras behave like an ordered rewrite system — closer to a formal grammar than to a natural-language description, which is why compiler theorists keep citing it.
Rick Briggs' proposal (1985)	A NASA researcher argued that Sanskrit's explicit case-marking and unambiguous syntax make it a promising representation format for knowledge in AI systems — a claim still debated, not settled.

Thread of Work	What It Involves
JNU's 'Sanskrit for AI' research	Computational linguists at Jawaharlal Nehru University and IIT Kanpur have built dependency parsers and sandhi-splitters that treat Panini's rules as an executable grammar.
Educational esolangs	Independent developers have built small teaching languages with Sanskrit-derived keywords, useful for pedagogy and cultural engagement rather than production software.

A handful of independent developers and student projects have built small teaching languages with Sanskrit-derived keywords — a loop called 'punः' (again), a conditional called 'yadi' (if) — in the same spirit as esoteric languages built around other scripts and idioms. These are genuinely fun and pedagogically useful for getting students interested in both linguistics and computer science, but they are teaching tools, not competitors to Python or Rust. A short illustrative sketch, not a working language, gives a sense of the idea:

```
यदि  (tāpamānam > 30)

  लिख  ("उष्णं अद्य")

अन्यथा

  लिख  ("शीतलं अद्य")
```

Translated: 'if temperature > 30, print "hot today", else print "cool today".' The value of exercises like this is educational and cultural, not computational — nobody would claim the Devanagari keywords make the logic run any differently than the English ones. The honest way to describe this whole thread of work is: a real and interesting research question about grammar formalism, wrapped in a much more speculative and often overstated popular narrative about Sanskrit as a secret weapon for AI.

5. What Tends to Get Left Out of the Optimistic Version

समस्याः

Enthusiasm for this field is easy to come by; follow-through is harder. A few honest caveats worth keeping in view:

- Annotated data is still the bottleneck, not model architecture. Bigger models trained on the same small, imperfect corpora don't fix the underlying scarcity.
- Script diversity means OCR progress on Devanagari doesn't automatically transfer to Grantha, Sharada, or Newari manuscripts — each needs its own training pass.
- Sandhi and compound-splitting still fail on genuinely ambiguous cases where trained Sanskrit scholars themselves disagree — a ceiling that more compute alone won't raise.
- The 'Sanskrit is perfect for AI' framing, when taken too literally, can crowd out the more modest and more useful engineering conversation about what specific problems this grammar formalism actually helps with.

6. Machine Learning Application in Details

यन्त्राधिगमस्य विशेषप्रयोगाः

Beyond the broad categories above, several specific ML techniques hold particular promise for Sanskrit and IKS research:

- ❖ Speech Recognition for Sanskrit Pronunciation
- ❖ Text Classification of Ancient Scriptures
- ❖ Automated Sandhi & Samasa Analysis
- ❖ Sanskrit Question Answering Systems
- ❖ Knowledge Graph Construction from IKS
- ❖ Semantic Search across Classical Texts
- ❖ Sanskrit Conversational Assistants
- ❖ Named Entity Recognition in Itihasa Texts

Each of these applications requires carefully constructed training datasets, close collaboration between domain experts and AI engineers, and thoughtful attention to the unique structural features of Sanskrit. None of them is beyond reach; several are already being actively developed by research groups in India and abroad.

7. Recommendation for policy and practice

नीतिनिर्माणाय सुझावाः

Meaningful progress in this field will require concerted action from multiple stakeholders — educational institutions, government bodies, technology organisations, and individual researchers. The following measures are recommended as priority areas:

1. Develop and publicly release a national Sanskrit AI dataset — annotated corpora covering grammar, semantics, manuscript text, and spoken audio.

2. Establish AI and Sanskrit Research Centres at central Sanskrit universities, staffed by interdisciplinary teams of linguists, historians, and computer scientists.

3. Promote open-source development of Sanskrit language tools — parsers, translators, and OCR engines — that the global research community can contribute to and build upon.

4. Integrate Sanskrit and IKS modules into the AI and Data Science curriculum under the National Education Policy 2020 framework.

5. Create multilingual digital learning platforms that deliver Sanskrit instruction in regional languages, making it accessible across India's diverse linguistic landscape.

6. Fund competitive research grants specifically for Sanskrit NLP, knowledge graph construction, and intelligent tutoring systems.

8. Future Scope and Vision

भविष्यस्य दृष्टिकोणः

The trajectory of AI development suggests that, within the coming decade, it will be technically feasible to create systems that can engage in real-time Sanskrit dialogue, generate metrically accurate Sanskrit verse, produce scholarly commentary on classical texts, and answer questions drawn from the entire corpus of Sanskrit literature — from the Rigveda to the works of Abhinavagupta.

Such systems, developed thoughtfully and in partnership with traditional scholarship, will not replace the Sanskrit pandit or the Vedic scholar. Rather, they will amplify human expertise, making it possible for a single scholar's knowledge to reach millions of students, and for a lifetime of manuscript study to be complemented by computational tools that can scan ten thousand documents in the time it takes to read one.

India stands at a unique historical juncture: it possesses the oldest and richest textual tradition in the world, and it is simultaneously emerging as a global leader in information technology and AI research. The convergence of these two streams is not an accident — it is an opportunity that this generation must seize with vision, urgency, and intellectual seriousness.

9. Closing Thought

उपसंहारः

The most durable version of this story isn't about Sanskrit being uniquely destined for the AI era — that's a claim best made carefully, if at all. It's a more modest one: a very old, very precisely structured language is getting serious computational attention because scholars and engineers are finally sitting in the same room often enough to do the work properly. Manuscripts get digitised. Compounds get split. Students get better tools. None of it needs to be mystical to be worth doing well.

सर्वे भवन्तु सुखिनः — may all beings be well.

Further Reading

- Briggs, R. (1985). Knowledge Representation in Sanskrit and Artificial Intelligence. AI Magazine, 6(1).
- Huet, G. (2003). Towards Computational Processing of Sanskrit. Proceedings of ICON.
- Kulkarni, A. & Shukla, D. (2009). Sanskrit Morphological Analyser: Some Issues. Sanskrit Computational Linguistics Symposium.
- Ministry of Education, Government of India. National Education Policy 2020; Indian Knowledge Systems Division, Implementation Framework (2023).
- National Mission for Manuscripts — public documentation on manuscript holdings and digitisation progress.