Chandojñānam

A Sanskrit Meter Identification and Utilization System

Hrishikesh Terdalkar, Arnab Bhattacharya

18th World Sanskrit Conference, 2023

Department of Computer Science and Engineering,
Indian Institute of Technology Kanpur

Introduction

Sanskrit Prosody

Recite

योऽन्तः प्रविश्य मम वाचमिमां प्रसुप्तां	yo'ntaḥ praviśya mama vācamimāṃ prasuptāṃ
सञ्जीवयत्यखिलशक्तिधरः स्वधाम्ना।	sañjīvayatyakhilaśaktidharaḥ svadhāmnā|
अन्यांश्च हस्तचरणश्रवणत्वगादीन्	anyāṃśca hastacaraṇaśravaṇatvagādīn
प्राणान्नमो भगवते पुरुषाय तुभ्यम्॥	prāṇānnamo bhagavate puruṣāya tubhyam||

What is the chanda **in this verse?**
Ans: वसन्ततिलका (Vasantatilakā)

Recite

योऽन्तः प्रविश्य मम वाचं प्रसुप्तां	yo'ntaḥ praviśya mama vācaṃ prasuptāṃ

Why does it feel odd?

- Deviation from a known pattern
- How do we know these patterns?
  - Sanskrit Prosody!

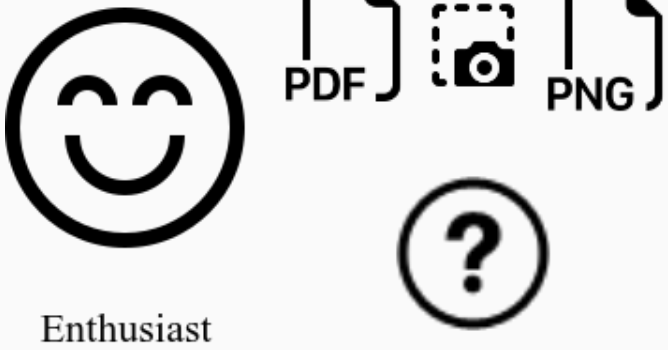
Background

- Classification of syllables
  - Pronunciation dependent
  - Laghu (*short*)
    - Letters with short vowels
  - Guru (*long*)
    - Letters with long vowels
    - Laghu letters followed by a joint letter (saṃyogaḥ)
    - Last letter of a pāda (conditional)
- Mātrā: Laghu 1, Guru 2
- Gaṇa: Sequence of three letters ($2^3 = 8$)

Chanda

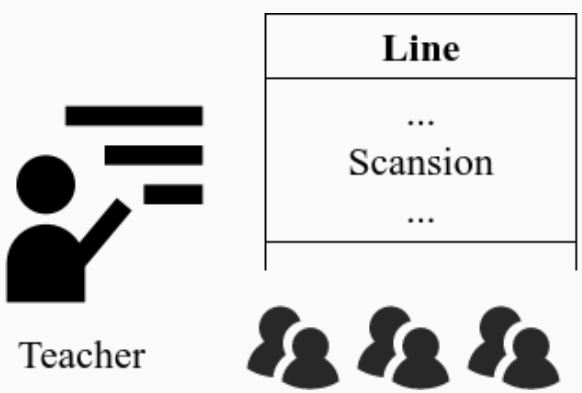
- Types
  - Akṣaracchanda: Sequences of laghu-guru
    - Samavṛtta, Ardhasamavṛtta and Viṣamavṛtta
  - Mātrācchanda: Counts of mātrā
- Literature: Vṛttaratnākaraḥ, Chandovicitiḥ, Chandomañjarī etc.

Motivation



Enthusiast

Poet



Teacher

Researcher

Why another meter identification tool?

Aim

- Add more user-friendly features.
- Catch errors in the text and suggest corrections!

- Web-based application
- Python library
- Three input modes: (1) plain text, (2) images (3) text files
- Two OCR Engines: (1) Google Drive OCR (2) Tesseract OCR
- Transliteration support (powered by `indic-transliteration`)
- Two meter identification modes: (1) line mode (2) verse mode
- Fuzzy matching support using edit distance comparison
- Informative scansion display
- Downloadable results

Feature Comparison

Features		[Mis07]	[MGS13]	[Raj20]	[Nei22]	Chandojñānam
Availability	Web Interface	✓[1]	✓[2]	✓	✓	✓
	Software Library			✓	✓	✓
Input	Text	✓	✓	✓	✓	✓
	Arbitrary Lines					✓
	Multiple Verses					✓
	Textfile Upload				✓	✓
	Image Upload					✓
Functionality	Meter Identification	✓	✓	✓	✓	✓
	Error Tolerance			✓	✓	✓
	Fuzzy Matching			✓		✓

Table 1: Feature comparison of extant meter identification systems

[1] `http://sanskrit.sai.uni-heidelberg.de/Chanda/HTML/` no longer functional.
[2] `https://sanskritlibrary.org:8080/MeterIdentification/` no longer functional.

System

Outline

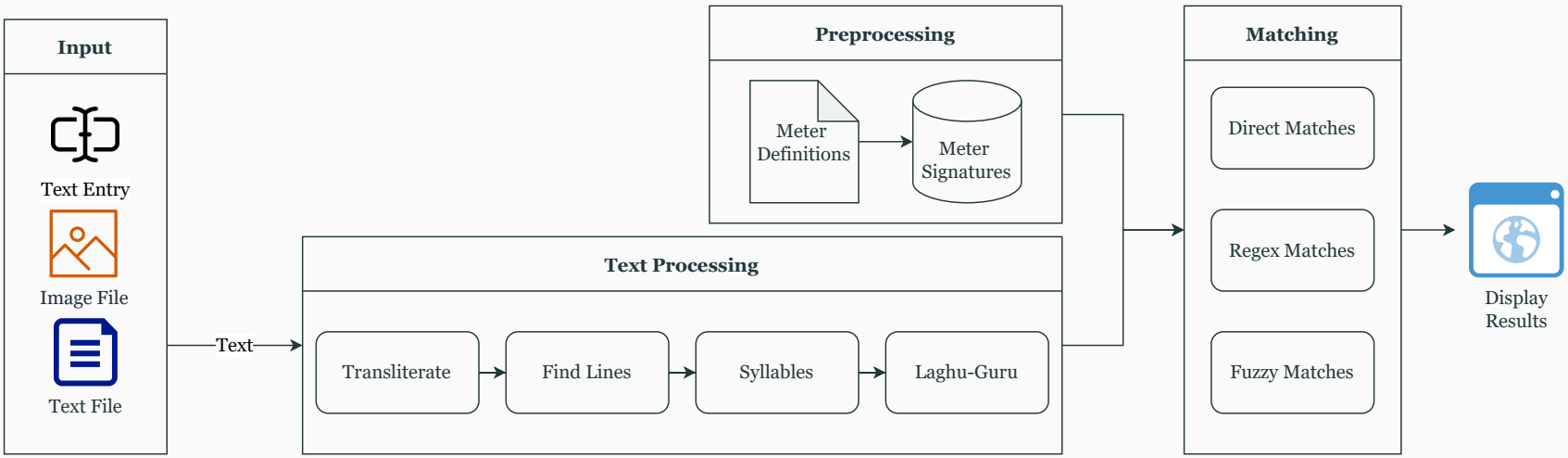


Figure 1: Workflow of Chandojñānam

Metrical Database

वृत्त	पाद	गण	लक्षण	अक्षरसङ्ख्या	मात्रा	यति
शार्दूलविक्रीडित		मसजसततग	गगगललगलगलललगगगलगगलग	19	30	12,7
शालिनी		मततगग	गगगगगलगगलगग	11	20	4,7
अपरवक्त्र	1	ननरलग	लललललललगलगलग	11	14	
अपरवक्त्र	2	नजजर	लललललगललगलगलग	12	16	
सौरभ	1	सजसल	ललगलगलललगल	10	13	
सौरभ	2	नसजग	लललललगलगलग	10	13	
सौरभ	3	रनभग	गलगलललगललग	10	14	
सौरभ	4	सजसजग	ललगलगलललगलगलग	13	18	
अनुष्टुभ्	1	----लग--	----लग--	8		
अनुष्टुभ्	2	----लगल-	----लगल-	8		

Figure 2: Generic Chanda definition format

- Column 'Pāda': index of pāda in the meter
- Uniform treatment of samavṛtta, ardhasamavṛtta and viṣamavṛtta
- Regex pattern (*regular expression*) definition

Metrical Database

- Two types of dictionaries
  - Signature of individual pādas (`CHANDA_SINGLE`)
  - Signature of consecutive pādas (`CHANDA_MULTIPLE`).

```
CHANDA_SINGLE = {
    'LGGLGGLGGLGG': ['Bhujaṅgaprayāta'],
    '[LG][LG][LG][LG]LG[LG][LG]': ['Anuṣṭubh (Pāda 1)'],
    '[LG][LG][LG][LG]LGL[LG]': ['Anuṣṭubh (Pāda 2)']
}

CHANDA_MULTIPLE = {
    'LGGLGGLGGLGGLGGLGGLGGLGG': ['Bhujaṅgaprayāta (Pāda 1-2)']
    '[LG][LG][LG][LG]LG[LG][LG][LG][LG][LG][LG]LGL[LG]': [
      'Anuṣṭubh (Pāda 1-2)'
    ]
}
```

Input

- Text input
- Textfile input
- Image file input

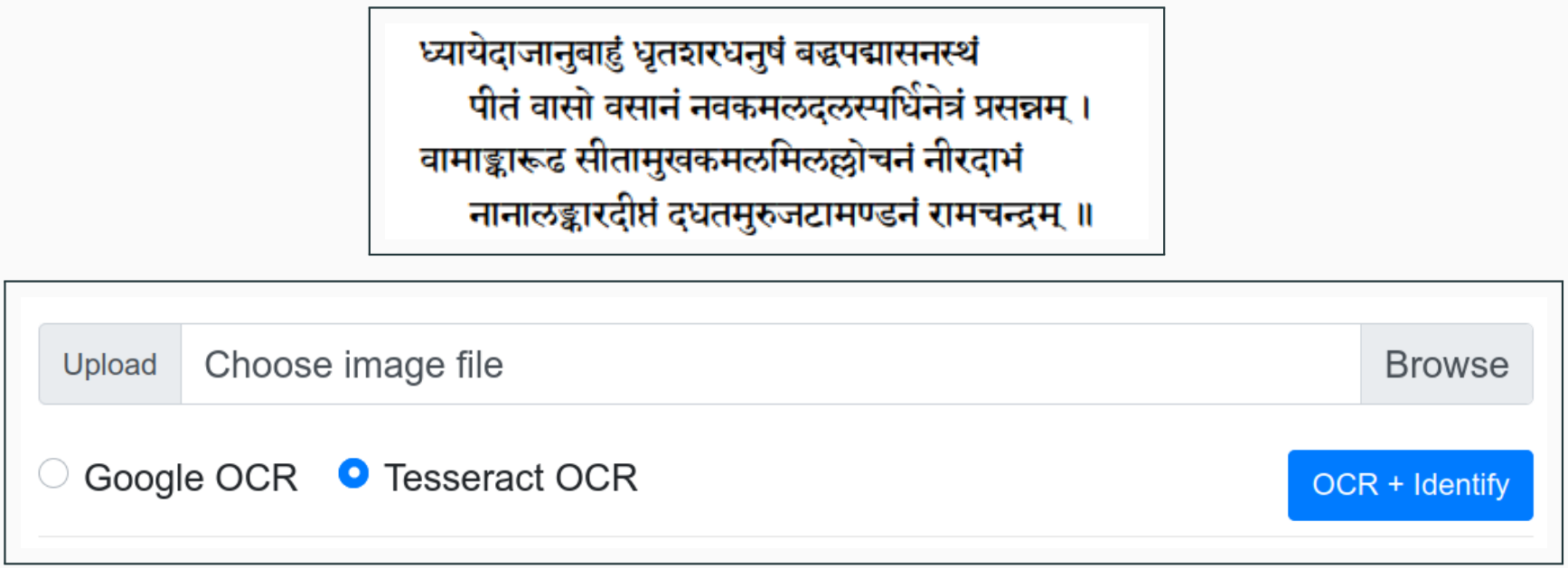


Figure 3: Upload a screenshot of a verse for meter identification

Text Processing

- Common text processing pipeline for all input modes
- Transliteration (*powered by* `indic-transliteration`[3])
  - Detect scheme
  - Convert to internal scheme (Devanagari)
- Line identification
  - Standard line-end markers: '\n', '।', '॥', '.'
- Syllabification (*powered by* `sanskrit-text`[4])
  - भारत = भा + र + त
- Laghu-Guru markers
  - Standard rules
  - ∃ Chanda where last letter should be laghu (Padānta Laghu)
  - Last letter not forced to be guru

[3] `https://pypi.org/project/indic-transliteration/`
[4] `https://pypi.org/project/sanskrit-text`

Meter Identification Algorithm

Algorithm 1: Meter Identification

Data: Metrical Database (MD)

Input: *lg-signatures* of every 'line' in the input ($T = \{lg_1, lg_2, \ldots, lg_n\}$)

Output: Result set containing exact or fuzzy matches

```
1  forall lg ∈ T do
2  |   SM_1 = FindDirectMatch(lg, 'CHANDA_SINGLE')
3  |   SM_2 = FindDirectMatch(lg, 'CHANDA_MULTIPLE')
4  |   RM = FindRegexMatch(lg, 'CHANDA_SINGLE' +
   |    'CHANDA_MULTIPLE')
5  |   DM = SM_1 + SM_2 + RM
6  |   FM = φ
7  |   if DM = φ then
8  |   |   FM = FindFuzzyMatch(lg)
9  |   end
10 |   return DM + FM
11 end
```

Direct Matching

Algorithm 2: Direct Matching

Input: *lg-signature*

Output: Result set containing exact matches

Function `FindDirectMatch`(*lg*, *'MD'*)

- $M_1 = \text{Query}(lg, \text{'MD'})$ `// dictionary lookup`
- $M_2 = \phi$
- **if** $M_1 = \phi$ **then** `// if no match found`
  - **if** *the last letter of lg is* laghu **then**
    - $lg_1 =$ *replace last letter of lg with* guru
    - $M_2 = \text{Query}(lg_1, \text{'MD'})$
  - **end**
- **end**
- **return** $M_1 + M_2$

Fuzzy Matching

What?

Finding approximate and close matches if an exact match is not found

Why?

- Digitally available Sanskrit text can be erroneous
  - Manual data entry
  - Post-scanning OCR followed by manual correction
- Types of Errors
  - Characters may be misspelt, e.g., रु (ru) as रू (rū)
  - Characters may be missing, e.g., वर्गै (vargai) as वगै (vagai)
  - Characters may be misidentified, e.g., ऋ (ṛ) as क्र (kra)
  - Characters may get split, e.g., ख (kha) as रव (rava)
- Several such errors can affect the metrical pattern of the text

Fuzzy Matching

How?

- **Problem**: Finding the *nearest matching string* for the *lg-signature* of the text
- Compute Levenshtein edit distance of the observed pattern (*powered by* `python-Levenshtein`[5])
- Normalize the edit distance by the length of target pattern

$$\text{Similarity} = 1 - \frac{\text{Levenshtein distance}}{\text{length of target match}}$$

- Topmost k matches as the possible fuzzy matches (currently, $k = 10$)
- Suggestions: changes required to transform the input into the target
  - **i**nsert, **d**elete, **r**eplace

[5] `https://pypi.org/project/python-Levenshtein/`

Fuzzy Matching Example – Matching

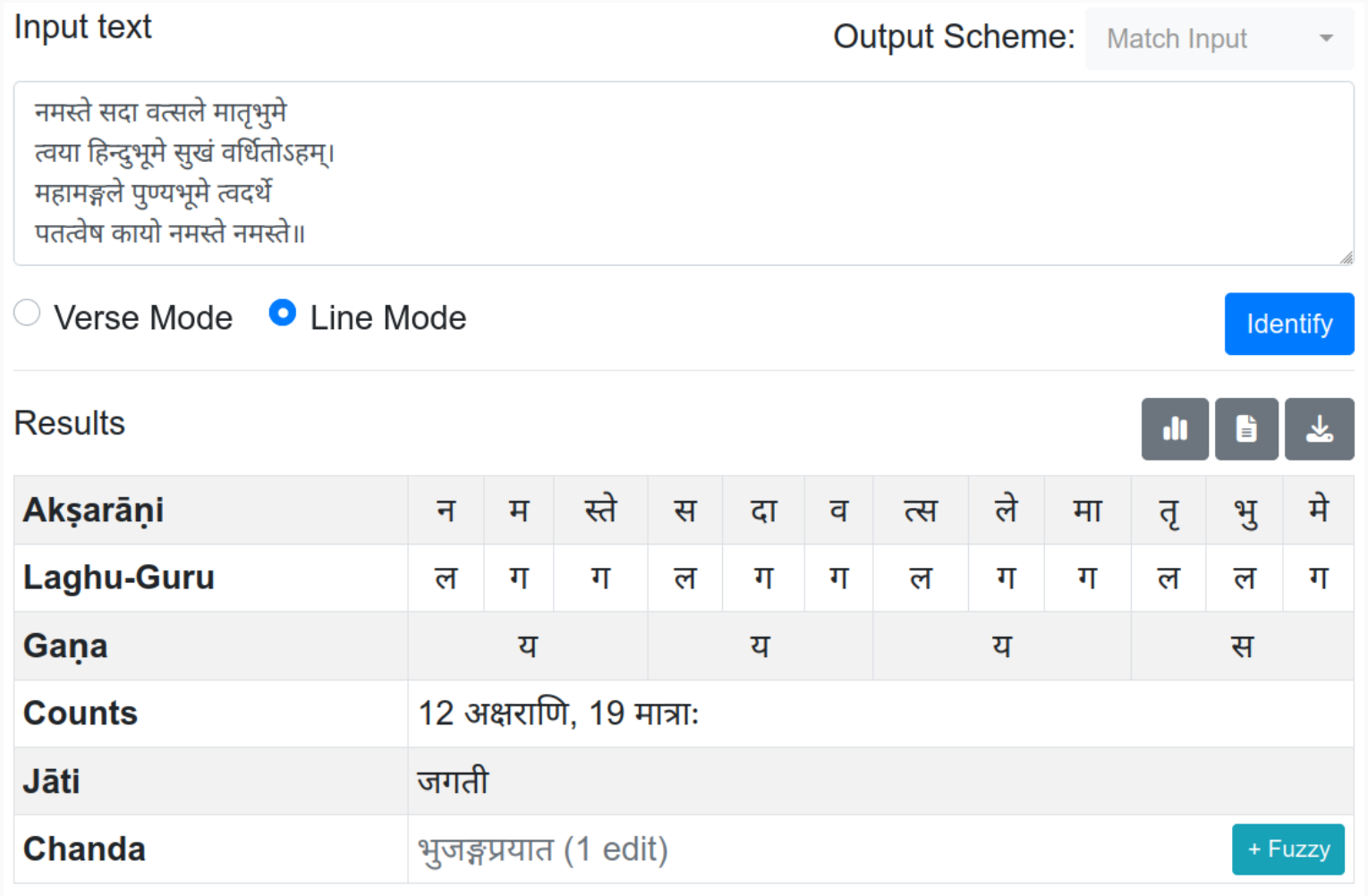


Figure 4: Meter identification with fuzzy matching

Fuzzy Matching Example – Suggestions

Chanda	भुजङ्गप्रयात (1 edit)	- Fuzzy

Fuzzy Matches

#	Chanda	Gaṇa	Cost	Similarity
1	भुजङ्गप्रयात	यययय	1	91.7%
	[[['न', 'म', 'स्ते'], ['स', 'दा'], ['व', 'त्स', 'ले'], ['मा', 'तृ', 'r(भु)[G]{भू}', 'मे']]]			
2	स्रग्विणी	रररर	2	83.3%
	[[['i(G)', 'd(न)', 'म', 'स्ते'], ['स', 'दा'], ['व', 'त्स', 'ले'], ['मा', 'तृ', 'भु', 'मे']]]			
3	विध्वङ्कमाला	ततगग	2	81.8%
	[[['d(न)', 'म', 'स्ते'], ['स', 'दा'], ['व', 'त्स', 'ले'], ['मा', 'तृ', 'r(भु)[G]{भू}', 'मे']]]			
4	हंसमाला (पाद 1-2)	सरभतगग	3	78.6%
	[[['न', 'i(L)', 'i(L)', 'स्ते'], ['स', 'दा'], ['व', 'त्स', 'ले'], ['मा', 'तृ', 'r(भु)[G]{भू}', 'मे']]]			
5	इन्द्रवंशा	ततजर	3	75.0%
	[[['d(न)', 'म', 'स्ते'], ['स', 'दा'], ['व', 'त्स', 'r(ले)[L]'], ['मा', 'तृ', 'i(G)', 'i(G)', 'मे']]]			

Figure 5: Fuzzy matching suggestions

Identification Modes

- *Line mode*: Treat the input as a set of arbitrary lines
  - Useful for checking meter of a single line or a set of unrelated lines
- *Verse Mode*: Treat the input as a collection of verses
  - Useful for identifying meter of a single or multiple verses
  - Utilizes information from other lines of the verse
  - Re-order results if required

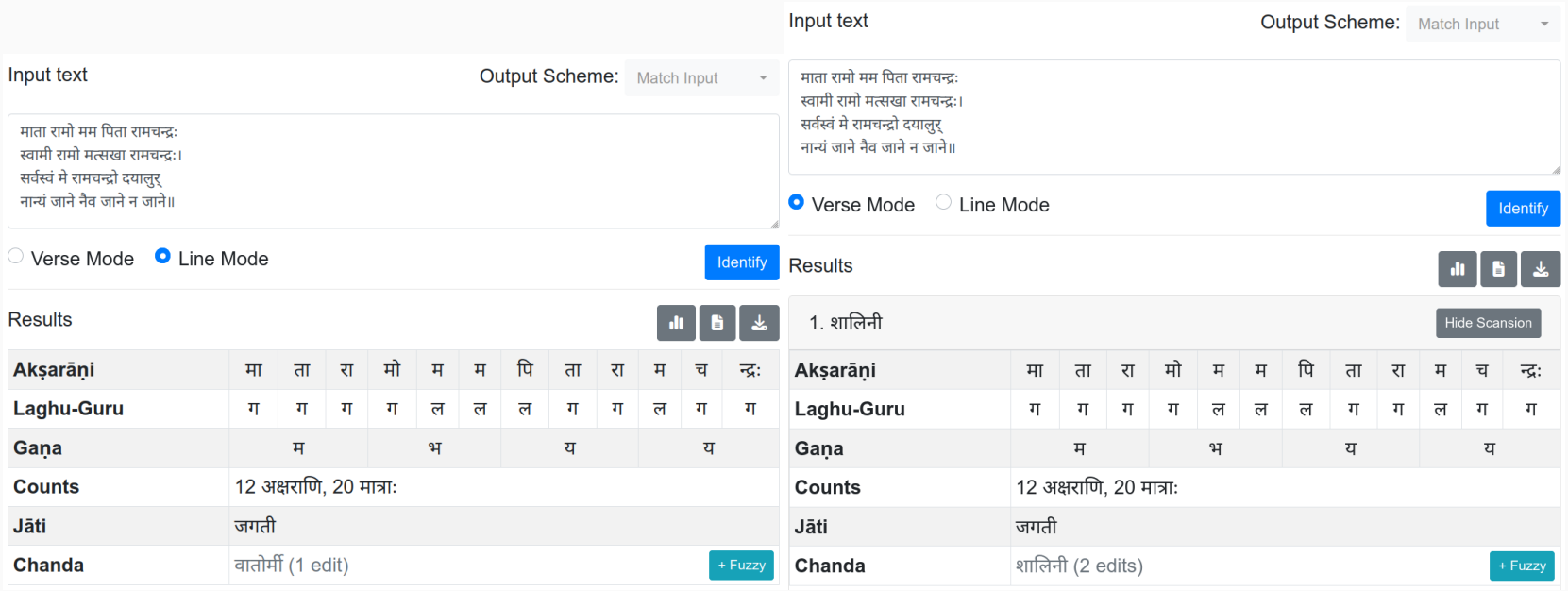


Figure 6: Meter identification in (a) *Line mode* and (b) *Verse mode*

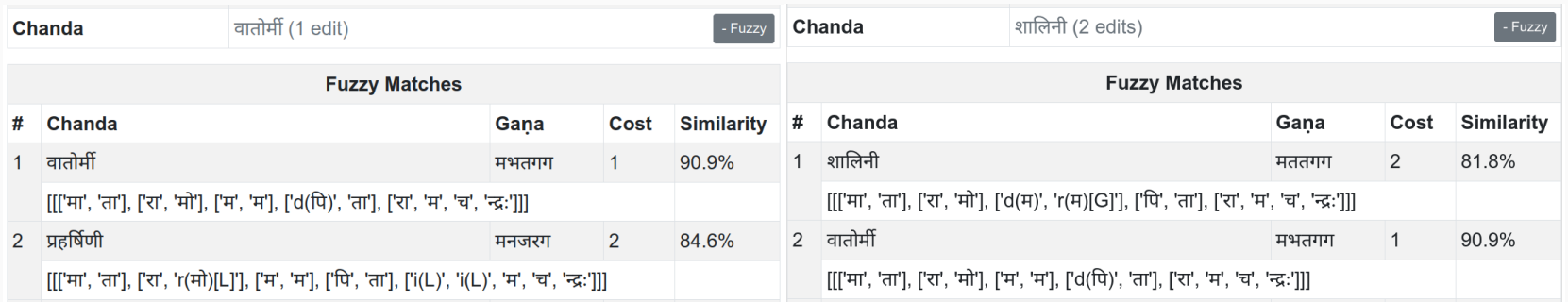


Figure 7: Fuzzy matches in (a) *Line mode* and (b) *Verse mode*

Other Languages

- Transliteration-based primitive[6] multilingual support

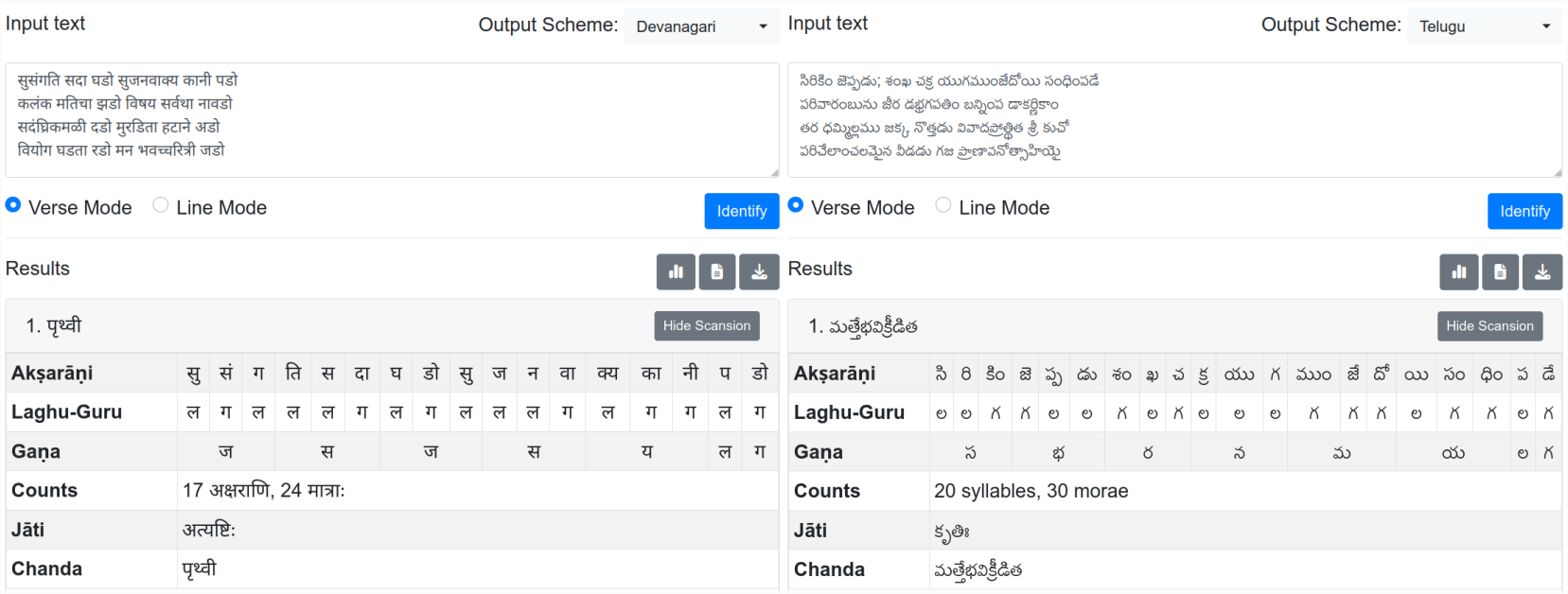


Figure 8: Identification from other Indian languages, (a) Marathi (b) Telugu

[6] Uses the rules and metrical database of Sanskrit prosody

Evaluation for Error Correction

Corpus

- Single text from different sources can differ in several places
- Three versions of Meghadūta[7] composed by Kālidāsa
  - Wikisource, sanskritdocuments.org and GRETIL
- Texts with more metrical variety
  - Śāntavilāsa (36 verses) (12 distinct meters)
  - Śrīrāmarakṣāstotra (39 verses) (9 distinct meters)
  - Rājendrakarṇapūra (72 verses) (4 distinct meters)
- Manually tagged meters for each verse from these texts
- Realistic evaluation: Simulate digitization pipeline for all four texts
  - Generate PDF from Wikisource text
  - Run both the OCR systems
  - Obtain the OCR-ed versions of the text
- Total 14 text versions, 1038 verses, exhibiting 17 distinct meters

[7] Also used by [Raj20] for evaluation

Results

		Meghadūta					Śāntavilāsa			Rāmarakṣā			Rājendrakarṇapūra			Total
		SD	GR	WS	GO	TO	WS	GO	TO	WS	GO	TO	WS	GO	TO	
Number of Verses		117	111	123	123	123	36	36	36	39	39	39	72	72	72	1038
Unique Chanda		1	1	1	1	1	12	12	12	9	9	9	4	4	4	17
Erroneous Verses		20	79	2	31	77	13	16	31	1	4	13	12	26	71	396
Correct	[Nei22]	20	79	2	30	66	11	13	14	0	2	9	12	24	36	318 (80.3%)
Meters	[Raj20]	19	79	2	30	75	12	15	24	1	2	9	12	26	58	364 (91.9%)
Identified	Chandojñānam	20	79	2	31	77	13	16	29	1	3	9	12	26	71	389 (98.2%)

Table 2: Error tolerance of meter identification systems. (Versions are WS: Wikisource, GO: Google OCR, TO: Tesseract OCR, SD: sanskritdocuments.org, GR: GRETIL.) Chandojñānam is able to detect correct chanda from erroneous verses 98.2% of the times.

Error Analysis

- Successfully detected and corrected two errors from Wikisource version of Meghadūta

Error #1

- Line: कालक्षेपं ककुभसुरभौ पर्वते पर्वेते ते (Pāda 3, Śloka 1.23)
- Incorrect word पर्वेते (should be पर्वते)
- Likely due to OCR error and an oversight by the curator
- Suggestion: [[['का', 'ल', 'क्षे', 'पं'], ['क', 'कु', 'भ', 'सु', 'र', 'भौ'], ['प', 'र्व', 'ते'], ['प', 'r(र्वै)[L]', 'ते'], ['ते']]]
- System correctly points to the location where a change is required

Error Analysis

Error #2

- Line: साभिज्ञानप्रहितकुशलैस्ततद्वचोभिर्ममापि (Pāda 3, Śloka 2.53)
- Extra letter (त) present in the sandhi of words कुशलैः and तद्वचोभिः
- Suggestion: [[['सा', 'भि', 'ज्ञा', 'न', 'प्र', 'हि', 'त', 'कु', 'श', 'लै', 'd(स्त)', 'त', 'द्व', 'चो', 'भि', 'र्म', 'मा', 'पि']]]
- Points out correctly that a syllable needs to be deleted
- However, points to an incorrect syllable स्त to be deleted
  - Both स्त and त are laghu letters
  - Deletion of either letter $\implies$ the correct metrical signature
  - Impossible for a meter identification based system

Conclusions

Conclusions and Future Work

- User-friendly meter identification system
- Several input options
- Focus on error detection and correction

Future Work

- Inclusion of Mātrācchandas
- Improvements to meter correction algorithm
- Possible consideration of semantic aspects
- More extensive support for Indian languages that use similar rules of prosody

Links

System: https://sanskrit.iitk.ac.in/jnanasangraha/chanda/
Source: https://github.com/hrishikeshrt/chanda/

References

References

[MGS13] Keshav S Melnad, Pawan Goyal, and Peter Scharf.
Meter identification of sanskrit verse.
The Sanskrit Library, USA, 2013.

[Mis07] Anand Mishra.
Sanskrit metre recognizer, 2007.

[Nei22] Tyler Neill.
Skrutable: Another step toward effective sanskrit meter identification.
2022.

[Raj20] Shreevatsa Rajagopalan.
A user-friendly tool for metrical analysis of sanskrit verse.
Computational Sanskrit & Digital Humanities, page 113, 2020.

Thank you!

Questions?