Unicode Normalization and Grapheme Parsing of Indic Languages

Nazmuddoha Ansary*, Quazi Adibur Rahman Adib*, Tahsin Reasat
Asif Shahriyar Sushmit, Ahmed Imtiaz Humayun, Sazia Mehnaz Kanij
Fatema, Mohammad Mamun Or Rashid, Farig Sadeque

Presented By:
Quazi Adibur Rahman Adib

* These authors share first authorship

LREC-COLING 2024 - The 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation

Introduction

- **Motivation**
  - Fixing Malformed Words
  - Encapsulating feature without disregarding writing forms of Indic Languages
- **Core Objectives**
  - Developing a Unicode normalizer for Indic languages to fix malformed words.
  - Developing a Grapheme Parser

Unicode Normalizer

- Broken Diacritics (BD)
- Broken Nukta (BN)
- Invalid Unicode (IU)
- Invalid Connector (IC)
- Fix Diacritics (FD)
- Vowel-Vowel Diacritic (VDV)
- Legacy Symbols Handling (LS)
- Language Specific Treatment

Broken Diacritics (BD)

Incorrect Form	Correct Form
সংস্কৃতি [স, ◌ং, স, ◌্, ক, ◌ৃ, ত, ◌ি] [U+09B8, U+0982, U+09B8, U+09CD, U+0995, **U+09C4**, U+09A4, U+09BF]	সংস্কৃতি [ʃɔŋskriṯi] [স, ◌ং, স, ◌্, ক, ◌ৃ, ত, ◌ি] [U+09B8, U+0982, U+09B8, U+09CD, U+0995, **U+09C3**, U+09A4, U+09BF]

Potential Source: Cognitive

Broken Nukta (BN)

Incorrect Form	Correct Form
কেন্দ্রীয়	কেন্দ্রীয় [kendrii̱o]
[ক, ে, ন, ্, দ, ্, র, ী, **য**, ়]	[ক, ে, ন, ্, দ, ্, র, ী, **য়**]
[U+0995, U+09C7, U+09A8, U+09CD, U+09A6, U+09CD, U+09B0, U+09C0, **U+09AF**, **U+09BC**]	[U+0995, U+09C7, U+09A8, U+09CD, U+09A6, U+09CD, U+09B0, U+09C0, **U+09DF**]

Potential Source: Keyboard

Invalid Unicode (IU)

Incorrect Form	Correct Form
াটোবাকো [া, ট, ো, ব, া, ক, ো] [**U+09BE**, U+099F, U+09CB, U+09AC, U+09BE, U+0995, U+0995, U+09CB]	টোবাকো [tobɐko] [ট, ো, ব, া, ক, ো] [U+099F, U+09CB, U+09AC, U+09BE, U+0995, U+0995, U+09CB]

Potential Source: Typing Mistake

Invalid Connector (IC)

Incorrect Form	Correct Form
দুইটি [দ, ু, ই, ্, ট, ি] [U+09A6, U+09C1, U+0987, **U+09CD,** U+099F, U+09BF]	দুইটি [d̪ʊ̯ɪ̯tɪ] [দ, ু, ই, ট, ি] [U+09A6, U+09C1, U+0987, U+099F, U+09BF]

Potential Source: Cognitive

Fix Diacritics (FD)

Incorrect Form	Correct Form
দুই [দ, ু, ু, ই] [U+09A6, U+09C1, **U+09C1,** U+0987]	দুই [d̪ʊ̄ɪ̄] [দ, ু, ই] [U+09A6, U+09C1, U+0987]

Potential Source: Keyboard

Vowel-Vowel Diacritic (VDV)

Incorrect Form	Correct Form
একএে [এ, ক, এ, ে] [U+098F, U+0995, **U+098F**, U+09C7]	একত্রে [ekotre] [এ, ক, ত, ্, র, ে] [U+098F, U+0995, **U+09A4**, **U+09CD**, **U+09B0**, U+09C7]

Potential Source: Typing Mistake and Cognitive

Legacy Symbols Handling (LS)

Incorrect Form	Correct Form
□ **[U+0980]**	৭ [ʃɐt̪] **[U+09ED]**

Potential Source: Typing Mistake

Language Specific Treatment [Assamese Replacement (AR)]

Incorrect Form	Correct Form
ব্যৱহাৰ [ব, ◌্, য, ৱ, হ, ◌া, ৰ] [U+09AC, U+09CD, U+09AF, **U+09F1**, U+09B9, U+09BE, **U+09F0**]	ব্যবহার [bæbohar] [ব, ◌্, য, ব, হ, ◌া, র] [U+09AC, U+09CD, U+09AF, **U+09AC**, U+09B9, U+09BE, **U+09B0**]

Potential Source: Cognitive

Language Specific Treatment [To-hosonto Normalize (THN)]

Incorrect Form	Correct Form
উত্স [উ, ত, ◌, স] [U+0989, **U+09A4**, **U+09CD**, U+09B8]	উৎস [utʃo] [উ, ৎ, স] [U+0989, **U+09CE**, U+09B8]

Potential Source: Cognitive

Language Specific Treatment [Complex Root Normalization (CRN)]

Incorrect Form	Correct Form
বি্ষ্পদ [ব, ি, ্, ষ, ্, প, ্, দ] [U+09AC, U+09BF, **U+09CD**, U+09B7, U+09CD, U+09AA, **U+09CD**, U+09A6]	বিষ্পদ [bɪʃpod̪] [ব, ি, ষ, ্, প, দ] [U+09AC, U+09BF, U+09B7, U+09CD, U+09AA, U+09A6]

Potential Source: Cognitive

Language	Total	Affected	%
Bangla	2,883,731	369,348	12.81
Devanagari	2,887,725	257,615	8.92
Gujarati	1,119,927	28,512	2.55
Odiya	414,483	36,267	8.69
Tamil	6,885,008	214,242	3.11
Punjabi	421,537	132,993	**31.55**
Malayalam	6,021,715	932,314	15.48

Table 1: Summary of the OSCAR Abugida language corpus. It shows how many, out of the total unique words, our pipeline was able to capture as unnormalized words

Word Level Experiments

NLP Experiments
[Normalization Performance]

Train Norm.	Test Norm.	F1 (NER)	F1 (Sentiment Analysis)
None	IndicNLP	0.89 ± 0.012	0.79 ± 0.014
	Ours	0.89 ± 0.011	0.79 ± 0.013
IndicNLP	None	0.90 ± 0.011	0.78 ± 0.002
	IndicNLP	0.91 ± 0.011	0.78 ± 0.001
Ours	None	0.90 ± 0.013	0.78 ± 0.008
	Ours	0.90 ± 0.007	0.78 ± 0.008

Table 2: Normalization Performance of IndicNLP (I) and our proposed (O) Normalizer.

NLP Experiments
[Robustness]

Train Norm.	Test Norm.	F1 (NER)	F1 (Sentiment Analysis)
None	att-1°	0.79 ± 0.039	0.77 ± 0.013
	att-5°	0.73 ± 0.011	0.74 ± 0.006
IndicNLP	att-1°	0.81 ± 0.003	0.76 ± 0.012
	att-5°	0.74 ± 0.013	0.73 ± 0.004
	att-1° + I	0.84 ± 0.019	0.76 ± 0.012
	att-5° + I	0.75 ± 0.015	0.73 ± 0.004
Ours	att-1°	0.80 ± 0.013	0.77 ± 0.004
	att-5°	0.73 ± 0.021	0.73 ± 0.013
	att-1° + O	0.89 ± 0.002	0.78 ± 0.008
	att-5° + O	0.89 ± 0.007	0.76 ± 0.007

Table 3: Robustness of IndicNLP (I) and our proposed (O) Normalizer. Here, att-{1, 5}° means attack degree.

Grapheme Parser

t = '''উজ্জ্বল নক্ষত্র অন্তোনীয় গ্রামসি'''

w = ['উ', 'জ', '্', 'জ', '্', 'ব', 'ল', ' ', 'ন', 'ক', '্', 'ষ', 'ত', '্', 'র', ' ', 'অ', 'ন', '্', 'ত', 'ো', 'ন', 'ী', 'য়', ' ', 'গ', '্', 'র', 'া', 'ম', 'স', 'ি'] n=32

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31]

C = [2,4,10,13,18,26]

Index: i , 0<i<n-1

Values: C[i=0]=2,C[i=1]=4

D = [(1,2,3),(3,4,5),(9,10,11),(12,13,14),(17,18,19),(25,26,27)] k = 6

Index: j, 0=<j=<k-1

Values:

D[j=0]=(1,2,3)> → (C[i=0]-1,C[i=0],C[i=0]+1)

D[j=1]=(3,4,5)> → (C[i=1]-1,C[i=1],C[i=1]+1)

Merging recursively

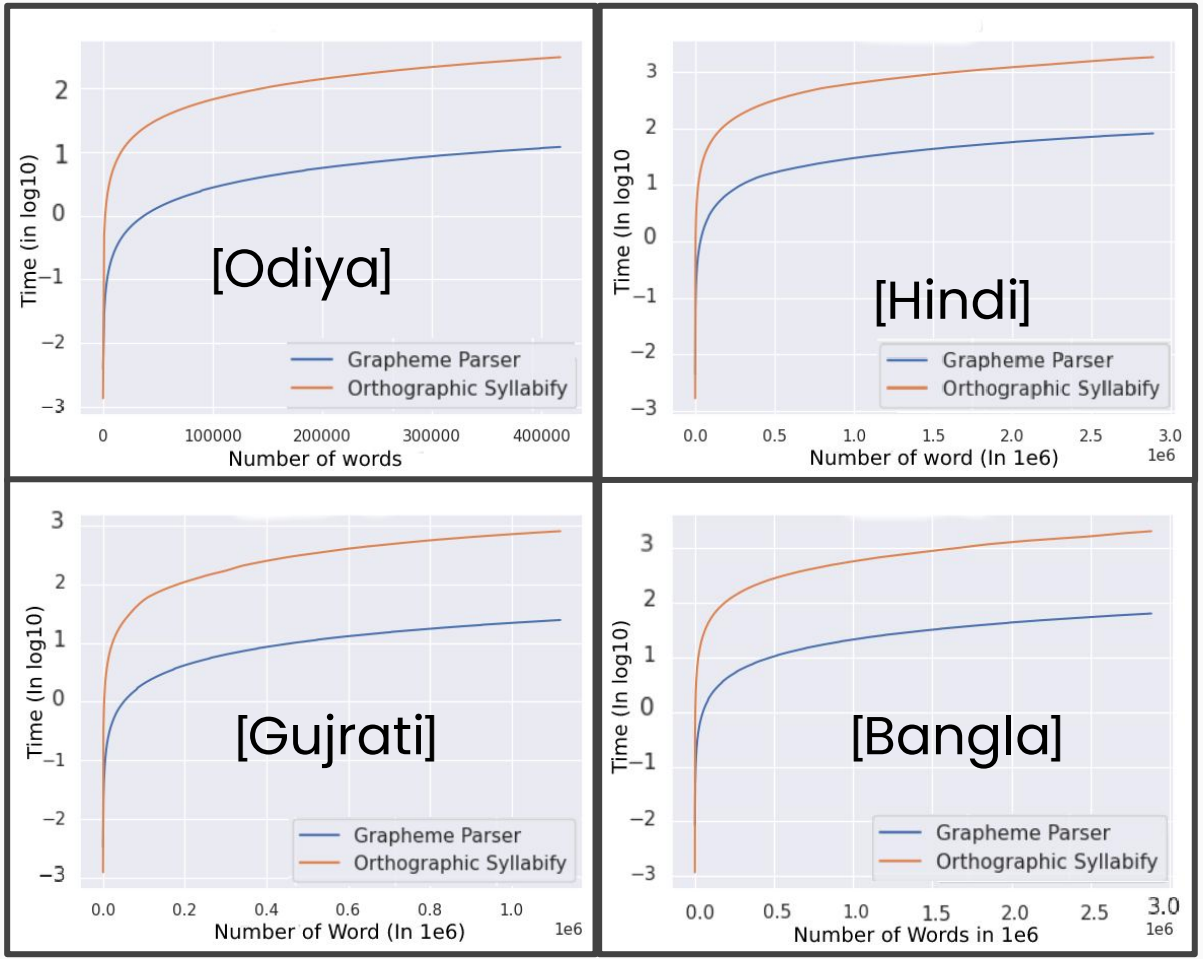
D' = [(1,2,3,4,5),(9,10,11),(12,13,14),(17,18,19),(25,26,27)] p = 5

w' = ['উ', 'জ্ব', 'ল', ' ', 'ন', 'ক্ষ', 'ত্র', ' ', 'অ', 'ন্ত', 'ো', 'ন', 'ী', 'য়', ' ', 'গ্র', 'া', 'ম', 'স', 'ি']

Merging gc , vd and cd

G = ['উ', 'জ্ব', 'ল', ' ', 'ন', 'ক্ষ', 'ত্র', ' ', 'অ', 'ন্তো', 'নী', 'য়', ' ', 'গ্রা', 'ম', 'সি']

Time taken for our parser to process examples vs. time taken by IndicNLP syllabifier. Time is shown as a logarithmic function, and the example count is in scale of 10^6.



Thank You

Grapheme Parser [Algorithm]

1. For a given text t, get the list of Unicode characters:
   w = u_0, u_1, u_2, ..., u_(n-1)

2. Create a list C with k elements having positions of connectors present in w in ascending order.
   Here k is the total number of connectors in w.
   For every element i in C: 0 < i < n-1 and u_(i-1), u_(i+1) are Consonants.

3. Create k lists D, maintaining the ascending order of C.
   Each list has 3 consecutive numbers.
   A list d that is an element of D is formed as: (i_j-1, i_j, i_j+1) where 0 <= j <= k-1.

4. Two lists are mergeable if and only if i_j+1 = i_(j+1)-1.
   The new merged list d in D will have 5 elements: (i_j-1, i_j, i_j+1, i_(j+1), i_(j+1)+1).
   Merge lists in D recursively until lists are not mergeable anymore, maintaining order in C.
   At the end, we have a new list D' with p lists, where each d' in D' has varying odd lengths >= 3.

5. For every d' with consecutive ascending numbers, merge the Unicode characters in w at those positions and construct w'.
   Elements in w' can only be: pure vowel diacritics vd, pure consonant diacritics cd, or grapheme root+conjunct diacritic gc.

6. To get the list of graphemes G, add a gc with the next element in w' if and only if the next element is vd or cd.