A Word-Splitting Approach to Kannada Sanskrit Sandhi Words Useful in Effective English Translation

Shanta Kallur [1], Basavaraj S. Anami [2]

[1]KLE Institute of Technology, Hubballi, Affiliated to
Visvesvaraya Technological University, Belagavi-590018, shanta.k@kleit.ac.in
[2]KLE Technological University, Hubballi, registrar@kletech.ac.in
Correspondence: shanta.k@kleit.ac.in

Abstract

Natural Language Processing is a branch of artificial intelligence that enables man-machine interactions through regional languages. In Kannada, there are two types of Sandhi: Kannada Sandhi and Sanskrit Sandhi. A morphphonemic word "Sandhi" is created when two words or distinct morphemes are joined or combined. Conversely, Sandhi word splitting reverses this process. Rules governing Sandhi exist across all the Dravidian languages. A rule-based method has been developed to split Sanskrit Sandhi words into their components within Kannada sentences. Once the Sanskrit Sandhi (SS) words are split, the type of Sandhi is also identified, facilitating accurate translation of the Sanskrit Sandhi words into English. This paper discusses seven types of SS words: SavarNadeergha, YaN, GuNa, Vruddhi, Jatva, Shchutva and Anunasika Sandhi. The identified split points adhere precisely to Sandhi rules. A dataset of 4900 Sanskrit Sandhi words found in Kannada sentences was used to evaluate the proposed method, which achieved an accuracy of 90.03% for Sanskrit Sandhi Identification and 85.87% for reliable English Translation. This work has potential applications in other Dravidian languages.

1 Introduction

Natural Language Processing (NLP) enables computers to understand any human spoken language in the real world, such as English, Hindi, Marathi, Tamil, Telugu, and Punjabi. NLP enables machines to comprehend human interactions. This involves identifying words within sentences based on word boundaries (Vempaty and Nagalla, 2011). Language facilitates communication among humans. Language grammar provides structure and is a system of rules that governs a language's correctness and adherence (Caryappa et al., 2020). Dravidian languages comprise a family of approximately 70 languages spoken by nearly 200 million people in various parts of India and the worldwide. In India, Tamil, Malayalam, Kannada, and Telugu, and over 20 non-literary languages are standard (Krishnamurthy, 2024). Kannada is a primary Dravidian language spoken mainly in Karnataka, with a rich cultural history spanning over 2500 years. It is the 27th most spoken language worldwide, with approximately 35 million speakers. Kannada faces challenges due to limited resources and significant syntactic and semantic variations. It has been less extensively studied in Machine Translation (MT) compared to other Indian languages (Nagaraj et al., 2021), making it more challenging task. Table 1 presents the number of speakers of Dravidian languages, categorized by state and globally.

Kannada has a linguistic construct called Sandhi (संधि in Sanskrit, ಸಂಧಿ in Kannada) where two words or morphemes merge, causing phonetic or morphological changes at the junction. This transformation is common in many Indian languages, including Sanskrit, Telugu, and Tamil, and is governed by specific grammatical rules. The word 'Sandhi' is used in both singular and plural forms throughout this paper. Splitting involves extracting the original words from the the Sandhi words and converting the Sandhi word into an equivalent English word (Natarajan and Charniak, 2011). Sandhi splitting approaches are generally classified into Dictionary-based, Rule-based, and Corpus-based methods (Shashirekha and Vanishree, 2016). There are two types of Sandhi in the Kannada Language: Kannada Sandhi and Sanskrit Sandhi (SS). Kannada Sandhi itself has three types: Lopa Sandhi, Aagama Sandhi, and Aadesh Sandhi. Sanskrit Sandhi includes seven types: SavarNadeergha Sandhi, GuNa Sandhi, YaN Sandhi, Vruddhi Sandhi, Jatva Sandhi, Shchutva Sandhi, and Anunasika Sandhi. The classification of Sandhi forms in Kannada is illustrated in Figure 1. This paper presents work on Sanskrit Sandhi, which supports the translation of Kannada texts

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 579–588

Language	Speakers	Locations
Telugu	83,000,000	Andhra Pradesh, Telangana and parts of Karnataka, UK, USA, Australia, Canada, UAE
Tamil	77,000,000	Tamil Nadu, Parts of Karnataka, Maharashtra, Kerala, France, Germany, Italy, USA, UK, Singapore & Srilanka
Kannada	45,000,000	Karnataka, Kerala, Tamil Nadu, Maharashtra, USA, UK, Canada, UAE, Saudi Arabia
Malayalam	37,000,000	Kerala, Tamil Nadu, Maharashtra, Karnataka
Tulu	1,850,000	Karnataka, Kerala, Gujarat, Saudi Arabia
Beary	1,500,000	Karnataka, Kerala, Gulf Countries
Brahui	2,430,000	Baluchistan (Pakistan), Helmand (Afghanistan)

Table 1: Speakers of Dravidian Languages

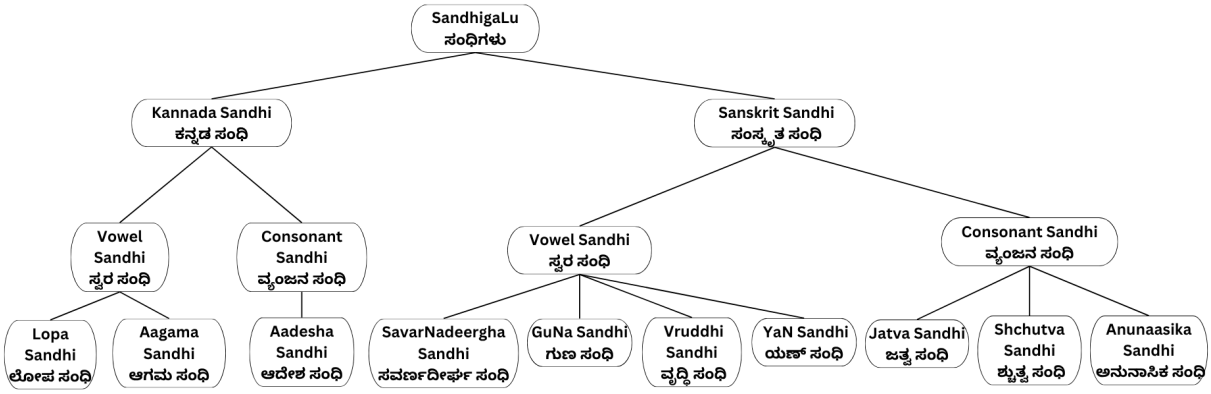

Figure 1: Classification of Sandhi Forms

into English, as a part of contribution to Machine Translation (MT).

MT bridges language barriers and is considered challenging for languages with complex linguistic structures, such as the Indian language Kannada. The challenges in MT are related to grammar, while others are related to language generation, multilingual dictionaries, word analysis, etc. (Alawneh and Sembok, 2011). Some existing translators, such as Google, Bing, Quillbot, and i-Translate, do not provide satisfactory translations of sentences containing Sandhi words. For example, the Kannada sentence ”ಯೋಗ ಮತ್ತು ಧ್ಯಾನ ಮನಶ್ಚಂಚಲತೆಯನ್ನು ಕಡಿಮೆ ಮಾಡುತ್ತದೆ” and its transliteration (TL) is ”Yoga mattu dhyana manaschaMchalatheyannu kaDime maDuttade”. Its English translation should be ‘Yoga and meditation reduce an unstable mind’. However, when we subjected this sentence to the existing translators, which failed to translate the given Kannada sentence, having the Sanskrit Shchutva Sandhi word ”ಮನಶ್ಚಂಚಲತೆ”, its transliteration (TL) form is ’manaschaMchalate’. Hence, the present paper proposes a rule-based Sandhi splitting method that is useful for converting Sanskrit Sandhi words to English, thereby effectively translating Kannada sentences into English.

2 Literature Survey

A literature survey was conducted to explore the current state-of-the-art methods for Sandhi splitting, identification, and machine translation.

Sandhi splitting in Tamil and Telugu has been modeled as a sequence-to-sequence (Seq2Seq) task using Transformer-based architectures. The study experimented with modeling at multiple granularities, including sentence-level, subword-level, and character-level representations, to effectively capture the morphological and phonological variations present in these languages (Dasari et al., 2025). Addressed the challenge of splitting Sanskrit Sandhi words where multiple phonetic transformarions obscure word boundaries. ABBIE, a new Attention-Based Bi-Encoder model that predicted the exact split point in Sanskrit compound words (Ali et al., 2025). Information retrieval (IR) in languages with complex morphological patterns, such as Indian languages, requires breaking down compound words (also called de-compounding)

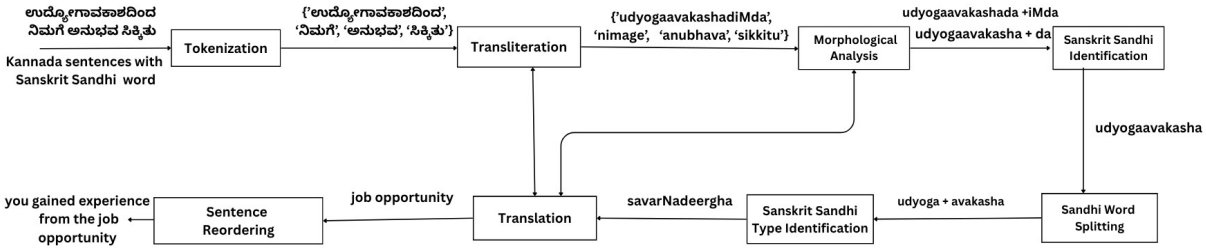

Figure 2: Block Schematic Diagram of Proposed Methodology

$SW = SW_1SW_2$, where $SW_1 = C_1C_2C_3C_4...C_n$ & $SW_2 = K_1K_2K_3K_4...K_n$

$SW = C_1C_2C_3C_4...CnK_1K_2K_3K_4....K_n$

Box 1: Structure of Sanskrit Sandhi Word

into their constituent parts. The corpus-based models were extensively used for de-compounding, requiring subtle assistance of semantics and managing sparsity (Sahu and Pal, 2024). Machine learning models were implemented using recurrent neural networks, long-short-term memory models, and double decoder models (S. et al., 2024). The morphological analysis of Sanskrit Sandhi words was context-dependent, and Sandhi split, also known as "Viccheda", was a challenging task. The existing methods included predetermined splitting rules. However, finding the exact split point was crucial as it determined the accuracy of constituent words (Madhura and Patankar, 2023). Nine methods were used for the "Sandhi" Splitter: the Bayesian Word Segmentation method, Conditional Random Field, Recurrent Neural Network, Hidden Markov Model, Rule-Based Approach (RBA), Deep Learning, Machine Learning, and Finite State Automata. Researchers developed Sandhi splitters using RBA (Gaikwad and Saini, 2021). Recurrent Neural Networks (RNNs) are widely used for machine translation. A combination of Naive Bayes and LSI (Latent Semantic Indexing) predicted the next word in Kannada translation. The model was trained using a variety of patterns created by combining bigram, trigrams, and 4-grams to enhance accuracy (Nandini et al., 2020). A seq2seq and LSTM models were used for Kannada Language to capture linguistic patterns in sentences (Nagaraj et al., 2021).The problem was a sequence-to-sequence prediction task and used modern deep-learning techniques. A compound-word (Sandhi) generation and splitting in the Sanskrit Language using LSTM and Bi-LSTM techniques were carried out, and a good prediction accuracy was achieved (Dave et al., 2021). The use of data and grammatical rules of Sanskrit played a significant role in splitting Upasarga and Pratyaya (Angle et al., 2018). The end-to-end neural network models resolved phonetic merges and broke compounds together to tokenize Sanskrit Sandhi. The character-level recurrent and convolutional neural networks helped to segment the words in Sanskrit (Hellwig and Nehrdich, 2018). The study specifically analysed the impact of Sandhi on building shallow parsers for Malayalam. Shallow parsing aims to identify correlated groups of words (chunks) in a sentence, serving as a key step in syntactic analysis (Devadath and Sharma, 2016).

The literature survey reveals that few authors have worked on Sandhi splitting for languages such as Telugu, Tamil, Sanskrit, Malayalam, and Kannada. Not all Sandhis are considered, and works have emphasised one or two types of Kannada Sandhi. Sanskrit Sandhi words in Kannada have not been explored extensively. Hence, this paper presents an account of translating Sanskrit Sandhi words into English from the Kannada language. It is a comprehensive work that encompasses all types of SS and the rules for splitting them into constituent words.

3 Dataset Preparation and Proposed Methodology

The necessary dataset has been collected and prepared for testing the method. The block schematic diagram of the processing stages is discussed.

3.1 Dataset Preparation

The data is collected from some Kannada storybooks and input from native Kannada language speakers, Kuvempu (1971) and Keshiraja (1920). The dataset comprises 4,900 Sanskrit Sandhi words drawn from Kannada sentences containing one, two, or three words of Sanskrit Sandhi, as listed in Table 2.

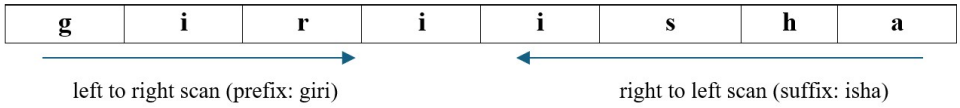

Figure 3: Prefix-Suffix method

Sentences	Count
Total No. of Sanskrit Sandhi words	4900
Total No. of Kannada Sentences	3736
No. of sentences without Sandhi words	39
Sentences having one Sandhi word	2768
Sentences having two Sandhi words	655
Sentences having three Sandhi words	274

Table 2: Sandhi Dataset

3.2 Proposed Methodology

The proposed methodology is divided into eight phases: Tokenization, Transliteration, Morphological Analysis, Sanskrit Sandhi Identification, Sandhi Word Splitting, Sanskrit Sandhi Type Identification, Translation, and Sentence Reordering, as shown in Figure 2. (Dasari et al., 2025)

3.2.1 Tokenization and Transliteration

The sentences are initially tokenized. It splits a given sentence into smaller units called tokens. Transliteration (TL): It is a phonetic method of writing, converting words from one language script to another by placing them in a familiar alphabet. Romanization transliterates the vowels and consonants of Kannada, as given in Tables 3 and 4, respectively. Transliteration changes the characters from the word's original alphabet to similar-sounding characters in a different script.

3.2.2 Morphological Analysis

Morphological analysis is used to identify all the morphemes from agglutinative words and their grammatical categories. This helps to improve the understanding of a language's word structure and meaning. Morphological analysis enables the accurate identification and reconstruction of original Sandhi words. Morphological analysis is crucial for Machine Translation (MT) and improves translation accuracy, especially for morphologically rich languages like Kannada. Since Kannada words often contain complex prefixes, suffixes, and Sandhi combinations, breaking them down correctly helps in meaningful translation into English or other languages.

3.2.3 Sanskrit Sandhi Identification

The obtained tokens are checked against a dictionary of root words to determine whether the token is a Sandhi word. The Sandhi words are identified based on their transformations. Let SW be the given Sanskrit Sandhi word, which is the concatenation of two words, namely SW_1 and SW_2, represented as SW=SW_1SW_2 where SW_1 and SW_2 are the two constituent words with sequences of characters as defined by expressions (1) and (2).

$$SW_1 = C_1C_2C_3C_4 \cdots C_n \quad (1)$$

$$SW_2 = K_1K_2K_3K_4 \cdots K_n \quad (2)$$

Let C_i and K_i represent the i^{th} character in words SW_1 and SW_2, respectively, and i = 1,2,3....n describe the characters in the words SW_1 and SW_2. The word SW can be written as shown in Box 1.

3.2.4 Sandhi Word Splitting

Sandhi Word Splitting (SWS), also known as Sandhi 'Viccheda', is a technique to split a string of conjoined words into a sequence of constituent root words. We have maintained dictionaries of prefixes, suffixes, and root words in a DWAG (Directed Word Acyclic Graph) structure. We have used prefix-suffix method for splitting Sandhi words. In the proposed prefix-suffix Sandhi splitting method, the Sandhi word undergoes character by character scanning, in both directions, resulting in prefix and suffix words. The SWS involves scanning from left to right to identify the prefix word, which is further verified against a dictionary, and from right to left to determine the suffix word, which is subsequently validated using the suffix dictionary, as shown in Figure 3. For example, the split of the word "ಗಿರೀಶ" (TL: giriisha) is shown in Figure 3. The given word will be spilt as ಗಿರೀಶ (TL: giriisha) –> ಗಿರಿ (TL: giri) + ಈಶ (TL: isha) by scanning from left to right and right to left, respectively.

3.2.5 Sanskrit Sandhi Type Identification

The Sandhi words are split, and the rules are applied to find the category of a Sandhi. The Sandhi

KV	TL	KV	TL	KV	TL	KV	TL	KV	TL	KV	TL
ಅ	a	ಆ	aa,A	ಇ	i	ಈ	ee, I, ii	ಊ	u	ಊ	oo, U, uu
ಋ	Ru	ಎ	e	ಏ	ai, ei	ಒ	o	ಓ	O	ಔ	au, ou

Table 3: Romanization of Kannada Vowels

KC	TL	KC	TL	KC	TL	KC	TL	KC	TL
ಕ	ka, qa	ಚ	ca, cha	ಟ	Ta	ತ	ta	ಪ	pa, fa, pha
ಖ	Ka, kha	ಛ	Ca	ಠ	Tha	ಥ	tha	ಫ	Pa
ಗ	ga	ಜ	ja	ಡ	Da	ದ	da	ಬ	ba
ಘ	Ga	ಝ	Ja, jha	ಢ	Dha	ಧ	dha	ಭ	Ba, bha
ಙ	ga	ಞ	ja	ಣ	Na	ನ	na	ಮ	ma
ಯ	ya	ರ	ra	ಲ	la	ಳ	La	ವ	va, wa
ಶ	Sa	ಷ	Sha	ಸ	sa	ಹ	ha		

Table 4: Romanization of Kannada Consonants

word is valid if it can be split into a prefix and a suffix. It is possible to identify the Sandhi split point by applying Kannada grammar rules, and the category of Sandhi (Aralikatte et al., 2018); (Gopal Krishna Udupa N, 2020).

i. Sanskrit Sandhi Rules There are seven types of Sanskrit Sandhi in Kannada and each Sandhi is governed by definite rule for joining the two constituent words. Following are the devised rules for governing Sanskrit Sandhi.

Rules	Split Words	Sandhi Word
a + a -> aa	deva + asura	devaasura
ಅ + ಅ ->ಆ	ದೇವ + ಅಸುರ	ದೇವಾಸುರ
aa + a -> aa	vidyaa + abhyasa	vidyaabhyasa
ಆ + ಅ -> ಆ	ವಿದ್ಯಾ + ಅಭ್ಯಾಸ	ವಿದ್ಯಾಭ್ಯಾಸ
i + i -> ii	kavi + iMdra	kaviiMdra
ಇ + ಇ -> ಈ	ಕವಿ + ಇಂದ್ರ	ಕವೀಂದ್ರ
u + u -> uu	vadhu + upadesha	vadhuupadesha
ಉ + ಉ -> ಊ	ವಧು + ಉಪದೇಶ	ವಧೂಪದೇಶ
i + ii -> ii	giri + iisha	giriisha
ಇ + ಈ ->ಈ	ಗಿರಿ + ಈಶ	ಗಿರೀಶ

Table 5: SavarNadeergha Sandhi Rules with Examples

- **SavarNadeergha Sandhi Rules**: When two vowels appear consecutively in a word, a single long vowel replaces both sounds. This process is known as extended vowel conjugation. The rules and examples are provided in Table 5.

- **Vruddhi Sandhi Rules**: If the prefix ends with characters 'a', and 'aa', and the suffix begins with characters 'i', 'ai', or 'au', during the Sandhi word formation, these are replaced by 'ai' and 'au', respectively. The rules, along with sample examples, are presented in Table 6.

Rules	Split Words	Sandhi Word
a + i -> ai	loka + ikya	lokaikya
ಅ + ಏ -> ಐ	ಲೋಕ + ಏಕ್ಯ	ಲೋಕೈಕ್ಯ
aa + ai -> ai	vidyaa + aishwarya	vidyaishwarya
ಆ + ಐ -> ಐ	ವಿದ್ಯಾ + ಐಶ್ವರ್ಯ	ವಿದ್ಯೈಶ್ವರ್ಯ
a + au -> au	Ghana + audharya	Ghanaudharya
ಅ + ಔ ->ಔ	ಘನ + ಔಧಾರ್ಯ	ಘನೌದಾರ್ಯ
aa + au -> au	mahaa + audharya	mahaudharya
ಆ + ಔ ->ಔ	ಮಹಾ + ಔಧಾರ್ಯ	ಮಹೌಧಾರ್ಯ

Table 6: Vruddhi Sandhi Rules with Examples

- **GuNa Sandhi Rules**: If the prefix ends with characters 'a' and 'aa' and the suffix begins with characters 'i', 'u', and 'ru', then the letters 'E', 'O', and 'ar' will be replaced in the Sandhi formation. This is called as "GuNa Sandhi". The rules, along with sample examples, are presented in Table 7.

- **Jatva Sandhi Rules**: The consonants 'k', 'ch', 'T', 't', and 'p' at the end of the prefix word are replaced by the third consonants of the same class ('g', 'j', 'D', 'd', 'b') at the beginning of the suffix word. This is called as "Jatva Sandhi". The rules, along with sample examples, are shown in Table 8.

- **YaN Sandhi Rules**: When a Sandhi is formed and if the prefix ends with characters 'i', 'u', and 'ru', then the character 'y' replaces 'i', the character 'v' replaces 'u', and the character 'r'

replaces the character 'ru'. This is called as "YaN Sandhi". The rules with sample examples are given in Table 9.

- **Anunasika Sandhi Rules**: The consonants 'k', 't', 'T', and 'p' at the end of the prefix word will be replaced with 'gm', 'n', 'N', and 'm' in Sandhi formation. The rules along with sample examples, are given in Table 10.

- **Shchutva Sandhi Rules**: The prefix word has 's' or 't' as ending characters, and the suffix word has 'sha' and 'cha' as beginning characters; then these are replaced by 'sha', or 'shcha' and 'chh', respectively. This is called as "Shchutva Sandhi". The rules, along with sample examples, are presented in Table 11.

Rules	Split Words	Sandhi Word
a + i -> E	sura + iMdra	surEMdra
ಅ +ಇ -> ಏ	ಸುರ + ಇಂದ್ರ	ಸುರೇಂದ್ರ
aa + i -> E	dharaa + iMdra	dharEMdra
ಆ + ಇ -> ಏ	ಧರಾ + ಇಂದ್ರ	ಧರೇಂದ್ರ
a + u -> O	soorya + udaya	sooryOdaya
ಅ + ಉ ->ಓ	ಸೂರ್ಯ + ಉದಯ	ಸೂರ್ಯೋದಯ
a + ru -> ar	deva + rushi	devarshi
ಅ + ಋ ->ರ್	ದೇವ + ಋಷಿ	ದೇವರ್ಷಿ
aa + ru -> ar	mahaa + rushi	maharshi
ಆ + ಋ -> ರ್	ಮಹಾ + ಋಷಿ	ಮಹರ್ಷಿ

Table 7: GuNa Sandhi Rules with Examples

Rules	Split Words	Sandhi Word
k -> g	vak + iisha	vagiisha
ಕ -> ಗ	ವಾಕ್ + ಈಶ	ವಾಗೀಶ
ch -> j	ach + aadi	ajaadi
ಚ ->ಜ	ಅಚ್ + ಆದಿ	ಆಜಾದಿ
T ->D	viraaT + roopa	viraaDroopa
ಟ -> ಡ	ವಿರಾಟ್ + ರೂಪ	ವಿರಾಡ್ರೂಪ
t-> d	sat + uddesha	saduddesha
ತ -> ದ	ಸತ್ + ಉದ್ದೇಶ	ಸದುದ್ದೇಶ
p->b	ap+ja	abja
ಪ -> ಬ	ಅಪ್+ ಜ	ಅಬ್ಜ

Table 8: Jatva Sandhi Rules with Examples

3.2.6 Translation and Sentence Reordering

In Machine Translation (MT), four methods are deployed: Hybrid, Rule-Based, Neural, and Statistical. This is also true for Kannada and its English equivalent. We have developed a rule-based machine translation method in the proposed approach that uses specialised dictionaries and Kannada grammar. For example the sentence "ಅವನು ಗಿರೀಶ ಇರುತ್ತಾನೆ" (TL: avanu giriisha iruttane). In this example, the Sandhi word ಗಿರೀಶ (TL:giriish) is extracted and split. The prefix ' ಗಿರಿ ' (TL:giri) and suffix ' ಈಶ ' (TL: iisha) are obtained using the Sandhi splitting method. The meaning of 'giri' means "mountain" in English and the meaning of 'iisha' means "lord". After combining, the meaning of the entire Sandhi word is "Mountain lord". It is the name of lord Shiva in Hinduism.

In sentence reordering, each non-Sandhi word's meaning is obtained in English, whereas the Sandhi words need splitting for correct translation. The words in the English sentence are tagged with PoS and reordered according to the SVO structure. Hence, we obtain the effective English translation as "He is the mountain lord", referring to Lord Shiva. The overall Proposed methodology is given in Box 2.

Rules	Split Words	Sandhi Word
i + a -> ya	ati + avasara	atyavasara
ಇ + ಅ ->ಯ	ಅತಿ + ಅವಸರ	ಅತ್ಯವಸರ
i + aa -> yaa	jaati + aatita	jaatyaatita
ಇ + ಆ -> ಯಾ	ಜಾತಿ + ಆತೀತ	ಜಾತ್ಯಾತೀತ
i + u -> yu	prati + uttara	pratyuttara
ಇ + ಉ -> ಯು	ಪ್ರತಿ + ಉತ್ತರ	ಪ್ರತ್ಯುತ್ತರ
u + a -> va	manu + aadi	manvaadi
ಉ + ಅ -> ವ	ಮನು + ಆದಿ	ಮನ್ವಾದಿ
ru + a -> ra	pitru + aajne	pitraajne
ಋ + ಅ -> ರ	ಪಿತೃ + ಆಜ್ಞೆ	ಪಿತ್ರಾಜ್ಞೆ

Table 9: YaN Sandhi Rules with Examples

Proposed Methodology

Input: Sentences with Sanskrit Sandhi Words

Output: Category of the Sanskrit Sandhi word and the Equivalent English sentence

Begin

Step 1: Accept the sentences with Sanskrit Sandhi words.

Step 2: Tokenize and transliterate (TT) the given sentence.

Step 3: Perform morphological analysis.

Step 4: Check for the Sanskrit Sandhi words and obtain the number of Sandhi words.

Step 5: For the number of Sandhi words, do

Step 5.1: Split the obtained Sanskrit Sandhi word into the prefix word and the suffix word.

Step 5.2: Apply the rules to identify the Sanskrit Sandhi.

Step 5.3: Translate the Sanskrit Sandhi word into English and replace it in the TT sentence.

Step 6: Translate other words in the TT sentence.

Step 7: Reconstruct the sentence based on the SVO structure.

End

Box 2: Overall Proposed Methodology

Rules	Split Words	Sandhi Word
k -> gm	vaak + maya	vaagmaya
ಕ್ -> ಙ್ಮ	ವಾಕ್ + ಮಯ	ವಾಙ್ಮಯ
t -> n	chit + maya	chinmaya
ತ್ --> ನ	ಚಿತ್ + ಮಯ	ಚಿನ್ಮಯ
T -> N	shaT + maasa	shaNmaasa
ಟ್ --> ಣ	ಷಟ್ + ಮಾಸ	ಷಣ್ಮಾಸ
p -> m	ap + maya	ammaya
ಪ್ --> ಮ	ಅಪ್ + ಮಯ	ಅಮ್ಮಯ

Table 10: Anunasika Sandhi Rules with Examples

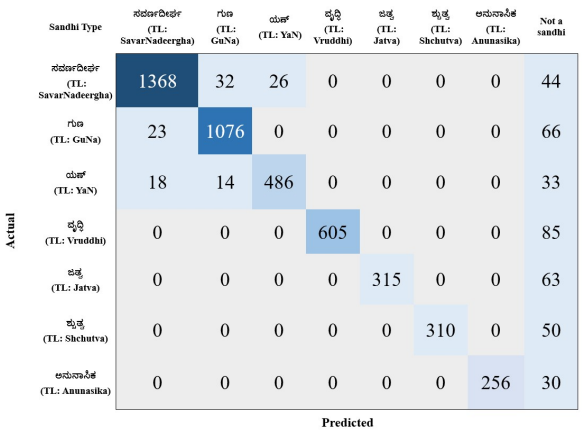

Figure 4: Confusion Matrix for Sanskrit Sandhi Splitting and Identification

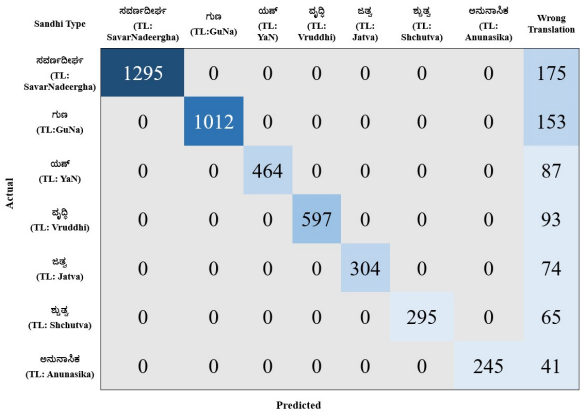

Figure 5: Confusion Matrix for Sanskrit Sandhi Translation

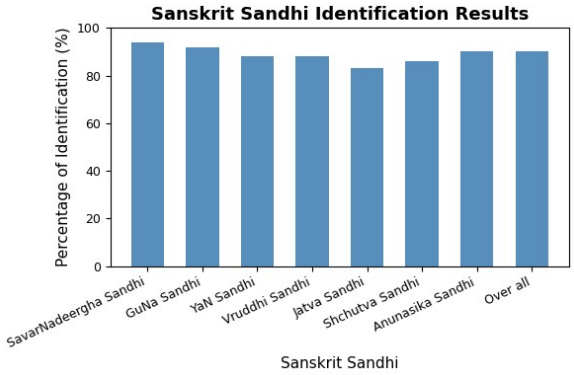

Figure 6: Sanskrit Sandhi Identification Results

4 Results of the Proposed Methodology

The proposed method is tested on a corpus of 3736 Kannada sentences containing 4900 Sandhi words, and the performance parameters are computed. The methodology is implemented in Python using the INLTK. The method's accuracy is defined as the average percentage of Sandhi words correctly identified (SI) and the percentage of Sandhi words correctly translated into English (ST).

$$\%\text{SIT} = \frac{\%\text{SI} + \%\text{ST}}{2} \quad (3)$$

A confusion matrix (CM) is obtained to determine how well the developed methodology compares with the desired or Actual outcomes. The CM for Sandhi identification and translation is shown in Figures 4 and 5. The performance parameters, including Precision, Recall, F1-score and Accuracy are presented in Tables 12 and 13. The accuracy of 90.03% (SI), 85.87% (ST), and 87.95% (SIT) is obtained for Sanskrit Sandhi identification and translation, respectively.

The accuracy graphs for all Sanskrit Sandhi identification and translation are shown in Figures 6 and 7, respectively. The BLEU score of 33 is achieved for translation using the Rule-based method. The proposed work is compared with existing works and is observed to be superior, as indicated in Table 14, by (Nagaraj et al., 2021), (Ali et al., 2025), and (Devadath and Sharma, 2016).The experiment is also conducted on an IndicBART a neural network model (Dabre et al., 2021) to corroborate the proposed RBM, that has given the Sandhi identification accuracy 88.3% and translation BLEU score 25 respectively.

4.1 Error Analysis

A single Sandhi word can have multiple valid split points, leading to identification errors. Prefix and suffix word ambiguities also contribute to errors, especially in guNa and vṛuddhi Sandhi. For example: ಮಹೋನ್ನತ (TL: mahonnata), ಮಹೌಧಾರ್ಯ (TL: mahaudhaarya) and ಮಹರ್ಷಿ (TL: maharshi). In the case of ಮಹರ್ಷಿ (TL:maharshi), the correct split is ಮಹಾ (TL:mahaa) + ಋಷಿ (TL:rushi), but the system may incorrectly split it as ಮಹ (TL: maha) + ಋಷಿ (TL:rushi). Such errors arise when words or morphemes are not

Rules	Split Words	Sandhi Word
s + sha -> sha	payas + shayana	payashayana
ಸ್ +ಶ ->ಶ	ಪಯಸ್ + ಶಯನ	ಪಯಶಯನ
s + cha -> shcha	manas + chaMchala	manashchaMchala
ಸ್ + ಚ -> ಶ್ಚ	ಮನಸ್ + ಚಂಚಲ	ಮನಶ್ಚಂಚಲ
th + cha -> chh	sharath + chaMdra	sharachhaMdra
ತ್ + ಚ -> ಚ್ಚ	ಶರತ್ + ಚಂದ್ರ	ಶರಚ್ಚಂದ್ರ

Table 11: Shchutva Sandhi Rules with Examples

Class Name	Recall	Precision	F1-score	Accuracy
ಸವರ್ಣದೀರ್ಘ ಸಂಧಿ (TL: SavarNadeergha Sandhi)|	0.93	0.97	0.95	0.95
ಗುಣ ಸಂಧಿ (TL: GuNa Sandhi)|	0.92	0.96	0.94	0.92
ಯಣ್ ಸಂಧಿ (TL: YaN Sandhi)	0.88	0.95	0.91	0.90
ವೃದ್ಧಿ ಸಂಧಿ (TL: Vruddhi Sandhi)	0.91	1	0.95	0.91
ಜಶ್ತ್ವ ಸಂಧಿ (TL:Jatva Sandhi)	0.83	1	0.91	0.84
ಶ್ಚುತ್ವ ಸಂಧಿ (TL: Shchutva Sandhi)|	0.89	1	0.93	0.86
ಅನುನಾಸಿಕ ಸಂಧಿ (TL: Anunasika Sandhi)|	1	1	1	0.90
Overall	0.91	0.98	0.94	0.90

Table 12: Sanskrit Sandhi Identification Performance Parameters

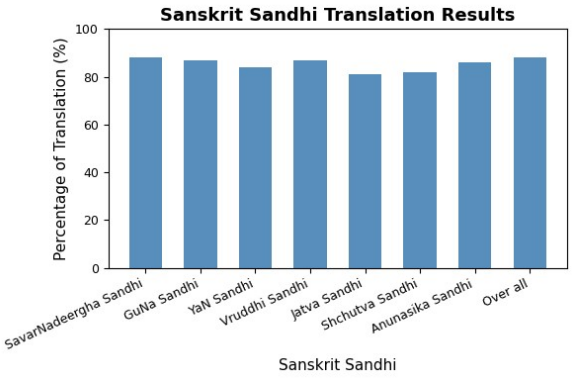

Figure 7: Sanskrit Sandhi Translation Results

present in the dictionaries. Thus, the accuracy of the rule-based method depends strongly on constantly updated prefix, suffix, and root word dictionaries.

In Split point identification, most of the incorrectly identified split points are character points between a word and an inflectional suffix attached to it. As the system evolves, this error gets automatically rectified by maintaining a finite list of inflectional suffixes in the language. The unidentified split points arise due to the specific rare patterns in Sandhi words.The rules devised are at the character-level and caused an unidentified split, which can be tackled with word-level information, such as POS tags.

5 Conclusion

The developed Rule-Based Methodology (RBM) for Sanskrit Sandhi splitting, identification, and English translation is tested on a corpus of 3736 Kannada sentences containing 4900 Sanskrit Sandhi words. It yielded satisfactory results for the Sanskrit Sandhi namely, SavarNadeergha Sandhi, GuNa Sandhi, YaN Sandhi, Vruddhi Sandhi, Jatva Sandhi, Shchutva Sandhi and Anunasika Sandhi. RBM has given an average accuracy of 90.03% for effective identification and 85.87% for translation of Sanskrit Sandhi words into English. It is observed that the accuracy of the RBM could be increased with the enhanced dataset and the corresponding prefix and suffix words dictionaries. INLTK Toolkit is used for implementation of the proposed methodology. There is scope to utilize statistical and deep learning-based methods, and the authors intended to explore them in future work. This methodology provides a valuable tool for Sandhi splitting in other Dravidian languages.

Limitations

The work focuses on all types of Sanskrit Sandhi words in Kannada sentences and their effective translations. The dataset and the dictionary sizes influence the performance and scope of improvements for both, aiming to enhance the accuracy of the proposed methodology.

Class Name	Recall	Precision	F1-score	Accuracy
ಸವರ್ಣದೀರ್ಘ ಸಂಧಿ (TL: SavarNadeergha Sandhi)	0.88	0.98	0.93	0.88
ಗುಣ ಸಂಧಿ (TL: GuNa Sandhi)	0.85	1	0.92	0.87
ಯಣ್ ಸಂಧಿ (TL: YaN Sandhi)	0.83	1	0.91	0.84
ವೃದ್ಧಿ ಸಂಧಿ (TL: Vruddhi Sandhi)	0.87	1	0.93	0.87
ಜಶ್ತ್ವ ಸಂಧಿ (TL:Jatva Sandhi)	0.80	1	0.89	0.81
ಶ್ಚುತ್ವ ಸಂಧಿ (TL: Shchutva Sandhi)	0.82	1	0.90	0.82
ಅನುನಾಸಿಕ ಸಂಧಿ (TL: Anunasika Sandhi)	0.86	1	0.92	0.86
Overall	0.84	0.99	0.91	0.85

Table 13: Sanskrit Sandhi Translation Performance Parameters

Paper Ref.	Language	Methodologies	Results	Remarks
(Dasari et al., 2025)	Tamil and Telugu	Seq2Seq and MT5	77.93%	Focused on the application of Sandhi rules. Illustrated word formation patterns and how lexical and functionalcategories together generate complex words
(Ali et al., 2025)	Sanskrit	Seq2Seq, LSTM, Bi-LSTM	89.27%	Proposed a deep learning framework using bi-encoders and a multi-head attention module to predict valid split points in Sanskrit compound words.
(Nagaraj et al., 2021)	Kannada	Seq2Seq , LSTM	86%	Captures linguistic patterns efficiently using RNNs limited by sentence /domain diversity.
(Dave et al., 2021)	Sanskrit	Seq2Seq, Deep learning methods	86.8%	Formulated Sandhi splitting as a sequence-to-sequence prediction task using deep learning.
(Hellwig and Nehrdich, 2018)	Sanskrit	Character-level RNN and CNN	85%	Contributed to low-resource language processing addressed morphological analysis for Sanskrit Sandhi word splitting.
(Devadath and Sharma, 2016)	Malayalam	Rule-based approach (RBA)	89%	A hybrid approach was employed, which first determines potential split points using statistical methods and subsequently segments the string into words by applying predefined character-level Sandhi rules.
Proposed Approach	Kannada	Rule-based (prefix-suffix, DWAG, comprehensive Sandhi rules)	90.03%	Works across all Sanskrit Sandhi types in the Kannada language, achieving higher accuracy with interpretable rules.

Table 14: Comparative Analysis with Existing Works

Acknowledgment

The Authors thank Dr.Vijayashree Hiremath for her support with Kannada Grammar and KLEIT and KLETechU for providing laboratory support.

Ethics Statement

The approach relies on linguistic rules and shows improved accuracy, possibly capturing context sensitive or culturally meaningful expressions. Caution should be exercised when working with religious or literary texts.The dataset used was created by authors and contains only Kannada words. We have carefully ensured that it includes no personally identifiable information or offensive content

References

Mouiad Fadiel Alawneh and Tengku Mohd Sembok. 2011. Rule-based and example-based machine translation from english to arabic. In *Proceedings - 2011 6th International Conference on Bio-Inspired Computing*, pages 343–47. Doi:10.1109/BICTA.2011.76.

I. Ali, L. Lo Presti, I. Spano, and M. La Cascia. 2025. Abbie: Attention-based bi-encoders for predicting where to split compound sanskrit words. In *International Conference on Agents and Artificial Intelligence*, page 334–344. Https://doi.org/10.5220/0013155300003890.

Sachi Angle, B. Ashwath Rao, and S. N. Muralikrishna. 2018. Kannada morpheme segmentation using machine learning. *International Journal of Engineering and Technology(UAE)*, 7(2):45–49. Doi:10.14419/IJET.V7I2.31.13395.

Rahul Aralikatte, Neelamadhav Gantayat, Naveen Panwar, Anush Sankaran, and Senthil Mani. 2018. Sanskrit sandhi splitting using seq2(seq)22. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018. Vol. 2*, pages 4909–14.

B. C. Caryappa, Vishwanath R. Hulipalled, and J. B. Simha. 2020. Kannada grammar checker using lstm neural network. In *Proceedings of the International Conference on Smart Technologies in Computing, Electrical and Electronics, ICSTCEE 2020*, pages 332–37. Doi:10.1109/ICSTCEE49637.2020.9277479.

Raj Dabre, Himani Shrotriya, Anoop Kunchukuttan, Ratish Puduppully, Mitesh M. Khapra, and Pratyush Kumar. 2021. Indicbart: A pre-trained model for natural language generation of indic languages. *Preprint*, arXiv:2109.02903.

P. Dasari, N. Vuppala, M. S. Gupta, P. Mishra, and P. Krishnamurthy. 2025. Sandhi splitting in tamil and telugu: A sequence-to-sequence approach leveraging transformer models. In *Proceedings - Inter-*

national Conference on Computational Linguistics, pages 93–103.

Sushant Dave, Arun Kumar Singh, A. P. Prathosh, and Brejesh Lall. 2021. Neural compound-word (sandhi) generation and splitting in sanskrit language.

V. V. Devadath and D. M. Sharma. 2016. Significance of an accurate sandhi-splitter in shallow parsing of dravidian languages. In *54th Annual Meeting of the Association for Computational Linguistics,*, pages 37–42.

Hema Gaikwad and Jatinderkumar R. Saini. 2021. On state-of-the-art of pos tagger, 'sandhi' splitter, 'alankaar' finder and 'samaas' finder for indoaryan and dravidian languages. *International Journal of Advanced Computer Science and Applications*, 12(4):429–36. Doi:10.14569/IJACSA.2021.0120455.

Gopal Krishna Udupa N. 2020. *Kannada Vyakarana Mattu Rachane*. Mcc Publications, 2016th ed.

Oliver Hellwig and Sebastian Nehrdich. 2018. Sanskrit word segmentation using character-level recurrent and convolutional neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018*, pages 2754–63. Doi:10.18653/v1/d18-1295.

Keshiraja. 1920. *Shabdamani Darpanam*. Karnataka Sahitya Parishat, Bengalore.

Bhadriraju Krishnamurthy. 2024. Dravidian languages | history, grammar, map, facts | britannica. *Retrieved*, 2024. Https://www.britannica.com/topic/Dravidian-languages.

Phadke Madhura and Shreya Patankar. 2023. Exploring the intricacies of sandhi in sanskrit: Phonological rules and linguistic significance. *International Journal of Applied Engineering Technology*, 5(1):353–60.

Pushpalatha Kadavigere Nagaraj, Kshamitha Shobha Ravikumar, Mydugolam Sreenivas Kasyap, Medhini Hullumakki Srinivas Murthy, and Jithin Paul. 2021. Kannada to english machine translation using deep neural network. *Ingenierie Des Systemes d'Information*, 26(1):123–27. Doi:10.18280/isi.260113.

B. R. Nandini, M. Hamsaveni, and V. Charunayana. 2020. Hybrid machine learning based kannada next word prediction. *International Research Journal of Engineering and Technology (IRJET)*, 7:5605–8.

Abhiram Natarajan and Eugene Charniak. 2011. S3 - statistical samdhi splitting. In *IJCNLP 2011 - Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 301–8.

H. S., Alok Nath M. Sreedeepa, Ajay K. Mani, C. Arun Kumar, and Sumam Mary Idicula. 2024. Review on sanskrit sandhi splitting using deep learning techniques. *Journal of Information Technology and Digital World*, 6(2):136–52. Doi:10.36548/jitdw.2024.2.003.

Siba Sankar Sahu and Sukomal Pal. 2024. A case study on decompounding in Indian language ir. *Natural Language Processing*, pages 1–31. Doi:10.1017/nlp.2024.16.

Vidwan N. Ranganath Sharma. 2010. *Vyakarana-Hosagannada*, 1 edition. Kannada Sahitya Parishat.

H. L. Shashirekha and K. S. Vanishree. 2016. Rule based kannada agama sandhi splitter. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 549–53. IEEE.

Phani Chaitanya Vempaty and Satish Chandra Prasad Nagalla. 2011. Automatic sandhi spliting method for telugu, an Indian language. *Procedia - Social and Behavioral Sciences*, 27(Pacling):218–25. Doi:10.1016/j.sbspro.2011.10.601.