Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Additional Diverse Techniques for Improvising Lesk Algorithm to Enhance Manipuri Word Sense Disambiguation

Chingakham Ponykumar Singh, H. Mamata Devi

Department of Computer Science, Manipur University, Manipur, India
Email: kleponymanipur@gmail.com

In the field of natural language processing (NLP), the essential task of understanding word meanings, known as Word Sense Disambiguation (WSD), is critical for many languages, including Manipuri. The ability to accurately interpret and differentiate word senses within sentences is crucial for effective communication. This paper presents a WSD system specifically designed for the Manipuri language, integrating diverse techniques with the Lesk algorithm to improve Manipuri WSD. The system particularly addresses the challenge of words with identical spellings by applying contextual analysis instead of relying solely on lemmatized forms. This method demonstrates superior performance compared to existing Manipuri WSD systems. Enhancements to the baseline Lesk algorithm are achieved through the incorporation of a Sense Inventory, IndoWordNet relationship data and Dictionary concepts. Additionally, the use of word relations, collocations, hand-coded rules,and identified keywords incrementally improves accuracy, with results ranging from 42.56% to 76.75%. All these diverse techniques drastically improve the word sense disambiguation for Manipuri language using Lesk language, which further give a challenge to the performance of the supervised techniques. All these techniques are tested on 1170 sentences which consist of ambiguous words. This research significantly advances the computational understanding and precision of Manipuri language processing, contributing to the broader field of NLP.

Keywords: Word Sense Disambiguation, Lesk Algorithm, Collocation, Hand coded rules, IndoWordNet.

1. Introduction

Language, as a dynamic system of communication, requires a deep understanding of subtle word meanings for effective interaction. In multilingual and diverse linguistic contexts like the Manipuri language, the task of WSD becomes crucial. Manipuri, a Tibeto-Burman language spoken in Manipur, employs two scripts: Bengali and Meitei/Meetei Mayek and is among the 22 official languages of India[14]. However, its complex morphology, numerous dialects and polysemy present unique challenges for natural language processing.

As with other languages, words in Manipuri can have different meanings based on their context. The same word can convey various situations or meanings depending on the area or topic, resulting in significant ambiguity. This ambiguity can hinder effective communication and the accurate representation of words. For instance, a word used in one region might have a completely different interpretation in another, complicating both human and machine understanding.

Word sense disambiguation offers a solution to these issues. WSD helps in resolving the ambiguities of word meanings by identifying the correct meaning of a word in context. This is vital not only for human comprehension but also for machine translation, where ambiguity remains a persistent challenge for researchers. Accurate WSD enhances the quality of machine translation, making it more reliable and understandable.

There are several approaches to WSD, each with distinct methodologies and applications. Some methods rely on supervised learning, utilizing annotated corpora to train models on word meanings. Others use unsupervised or semi-supervised techniques, leveraging large amounts of unannotated text to infer word senses based on context. Lexical databases, such as WordNet, also play a crucial role by providing structured information about word meanings and their relationships.

1.1 Knowledge-based approach

The knowledge-based approach to WSD relies on external lexical resources such as WordNet, dictionaries, thesauri, ontologies and collocations[10]. These resources provide a collection of information in a given field, such as medical diagnosis and may use techniques like overlap-based methods, grammatical rules, or hand-coded rules for disambiguation. The aim is to deduce insights from the information housed in these resources to infer the senses of words in context. While knowledge-based approaches generally have lower performance compared to other methods, they are particularly effective for agglutinative languages and in scenarios with limited resources. This approach often requires a machine-readable dictionary (MRD) to determine the different features of ambiguous words and their context.

1.2 Machine learning based approach

In the area of machine learning, systems are developed to execute WSD through a classifier that identifies unique features and assigns senses to new instances. Typically, the input includes the target word along with its surrounding context. In this method, the features are derived directly from the words themselves, with the feature values indicating the frequency of these words appearing near the target word. This technique is categorized into supervised, unsupervised and semi-supervised methods.

1.1.1 Supervised techniques: Machine learning techniques are utilized to develop a classifier using datasets that have been manually annotated to indicate different senses. This classifier, often referred to as a word expert, is dedicated to a specific word and assigns the appropriate meaning to each instance. The training dataset consists of examples where the target word has been manually tagged with a particular sense derived from a reference dictionary.

1.1.2 Unsupervised techniques: This approach eliminates the need for manually labelled sense information. Instead, it is predicated on the principle that words with analogous meanings will frequently appear in similar contexts. By analyzing the surrounding words, the method identifies distinct senses of a word. This is achieved by clustering instances of the word based on their context, effectively grouping occurrences that share common characteristics. The ultimate aim is to categorize new instances of the word into these pre-formed clusters, thereby accurately determining their sense based on contextual similarities. This clustering method leverages the natural tendency of words with similar meanings to share contextual patterns, providing an efficient and automated way to disambiguate word senses.

1.1.3 Semi-Supervised techniques: These techniques operate well even with minimal annotated data, often surpassing completely unsupervised methods, particularly with extensive datasets. They rely on a small amount of reference information and still achieve impressive results. In scenarios involving large volumes of data, these methods prove to be highly effective. Utilizing limited labeled data, they can perform better than entirely unsupervised approaches.

1.1.4 Hybrid approach: A hybrid method integrates aspects of both knowledge-based and machine-learning approaches to utilize their respective advantages. Achieving precise WSD is essential for various applications including question answering, machine translation and sentiment analysis[17].

Beside the above mentioned approaches many other methodologies have been adopted. Study on recent trends of WSD have shown that there is a great shift in the WSD methodologies like transformer based approach[31, 32].

For Manipuri, a language with scarce digital resources and limited NLP tools, developing efficient WSD mechanisms is especially important.

1.3 Data collection

Data is essential for research, forming the basis for conclusion and verification. Its role is crucial, supporting various vital research functions. This study utilizes the Manipuri Monolingual Corpus, Manipuri IndoWordNet data and the Manipuri Electronic dictionary data. The first two datasets are sourced from Technology Development for Indian Languages (TDIL), while the third is directly obtained from its developer.

1.3.1 Manipuri corpus (Monolingual corpus)

The database plays a pivotal role in compiling the dataset employed in this research. The accumulated data is presented in the Bengali script and spans a broad spectrum of fields, including science, arts, literature and media. Within the arts domain, the dataset incorporates information on a wide array of subjects such as economics, history, law, linguistics, philosophy, politics, psychology, religion and sociology. The scientific domain encompasses diverse subjects like biology, botany, chemistry, geography, mathematics, medicine, physics, wildlife and zoology.

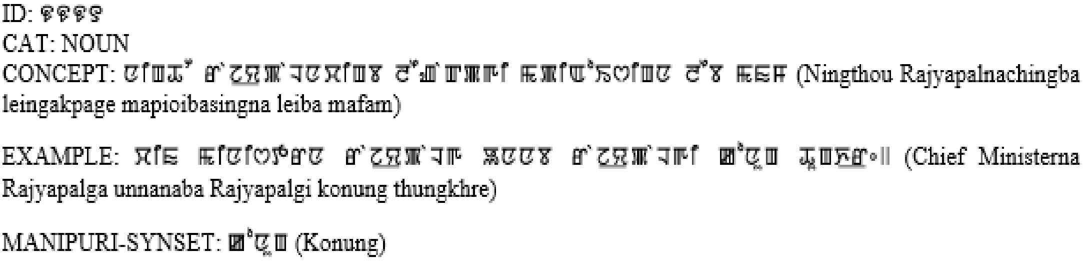
In the literature domain, the data covers an extensive range of areas, including arts and crafts, literary criticism, culture, didactic works, novels, short stories, theatre and trivia. Additionally, the dataset integrates information from magazines and newspapers, which fall under the media category. This comprehensive collection aims to illustrate the cultural and societal applications of the Manipuri language, providing a holistic view of its use.

The multifaceted nature of this data collection endeavours to explore every conceivable domain where the Manipuri language might be employed. It aims to uncover the various contexts and settings in which the Manipuri language is utilized, be it in newspapers, academic circles or other forms. This detailed exploration is intended to shed light on the diverse and rich applications of the language, reflecting its significance in different spheres of life.

1.3.2 IndoWordNet

IndoWordNet serves as an extensive lexical resource, covering 18 languages spoken in India. This project, developed through a collaborative approach, provides a detailed structure of various semantic relationships such as hypernyms, hyponyms, synonyms, antonyms and meronyms, crucial for exploring language semantics. By accommodating diverse linguistic variations, IndoWordNet significantly enriches our understanding of word senses in Manipuri and other languages[9,27,30]. It is instrumental in enhancing machine learning applications, ensuring precise language generation and comprehension. This essential repository boasts a lexicon comprising 16,323 entries, including 10,156 nouns, 3,806 adjectives, 332 adverbs and 2,021 verbs, forming a critical foundation for linguistic research[25].

Srtucture of IndoWordNet’s Manipuri data:



ID: ꯴꯴꯴꯵
CAT: NOUN
CONCEPT: ꯅꯤꯡꯊꯧ ꯔꯥꯖ꯭ꯌꯄꯥꯜꯅꯆꯤꯡꯕ ꯂꯩꯉꯥꯛꯄꯒꯦ ꯃꯄꯤꯑꯣꯏꯕꯁꯤꯡꯅ ꯂꯩꯕ ꯃꯐꯝ (Ningthou Rajyapalnachingba leingakpage mapioibasingna leiba mafam)

EXAMPLE: ꯆꯤꯐ ꯃꯤꯅꯤꯁ꯭ꯇꯔꯅ ꯔꯥꯖ꯭ꯌꯄꯥꯜꯒ ꯎꯅꯅꯕ ꯔꯥꯖ꯭ꯌꯄꯥꯜꯒꯤ ꯀꯣꯅꯨꯡ ꯊꯨꯡꯈꯔꯦ꯫ (Chief Ministerna Rajyapalga unnanaba Rajyapalgi konung thungkhre)

MANIPURI-SYNSET: ꯀꯣꯅꯨꯡ (Konung)

In this context, MANIPURI-SYNSET denotes a term (occasionally encompassing words with similar meanings) and its particulars are outlined in the aforementioned structure within IndoWordNet. ID serves as the exclusive identifier linked to a specific word, while CAT designates the part of speech associated with that word. CONCEPT encapsulates the significance or sense conveyed by the word and EXAMPLE presents a sentence incorporating the designated term.

The Manipuri language benefits significantly from the IndoWordNet data as a highly reliable source for the word sense disambiguation system (WSD). This particular dataset stands out due to its comprehensive coverage of essential WSD elements, encompassing word senses, synonyms, hypernyms, hyponyms and meronyms specific to Manipuri words.

There exists an opportunity for enhancement in various aspects of WordNet to align with the rigorous demands of Knowledge-based WSD systems[10]. Noteworthy adjustments can be made across different fields within WordNet, including but not limited to synonyms, hyponyms, hypernyms, meronyms, glosses, distributional constraints and frequently used glosses. Suggestions have been proposed to modify these fields, indicating their pivotal role in elevating the accuracy of knowledge-based WSD systems[21].

1.3.3 Electronic Dictionary

The electronic dictionary is a comprehensive collection of Manipuri vocabulary, including definitions and sample sentences. This extensive database, encompassing 10,337 entries, results from the scholarly efforts of Dr. S. Poireiton Meitei[29]. Authorization to utilize this information was secured from the original creators. This resource is an invaluable tool for exploring the lexical characteristics of Manipuri, offering crucial examples for contextual understanding.

The structure of the paper is outlined as follows: Section 2 offers a review of relevant literature, Section 3 explores into a thorough examination of the algorithm, Section 4 shows the analyzed results and Section 5 gives the conclusion of the paper.

2. Literature review

Languages, such as Manipuri, are abundant with words that carry multiple meanings, presenting a challenge where identical terms can imply different things depending on their usage in various sentences. Ambiguity in language can appear in several ways, including homonyms, which are words spelled the same but have different meanings and homophones, where words sound the same but have different spellings. This complexity highlights the need for disambiguation methods to ensure precise understanding in both natural language processing and everyday communication.

A. Zouaghi et al. [23] examine different variants of the Lesk algorithm for WSD in Arabic, utilizing the Arabic Wordnet (AWN). The study compares these variants with fifty highly ambiguous Arabic words and finds that the modified Lesk algorithm, incorporating AWN and various similarity measures, reaches a commendable precision of 67%. The Leacock and Chodorow measure shows the highest disambiguation performance among the tested measures. Manish Kumar et al. [12] developed a WSD system that resolves ambiguities by using contextual clues from surrounding words and extensive WordNet data. This innovative method converts raw input into a more analyzable format, enhanced with details such as Part of Speech (POS) tags and attributes related to subjects and objects. This enhancement aids the classifier in deciphering ambiguous words within sentences.

Basile et al. [11] propose a hybrid WSD approach combining knowledge-based and supervised learning techniques. This method addresses the challenge of disambiguating languages with limited training data and sense definitions, such as Italian. Experiments on the EVALITA 2007 dataset demonstrate that integrating knowledge-based and supervised learning methods yields the best performance, with the JIGSAW algorithm providing a promising solution for semantic ambiguity. Pooja Sharma and Nisheeth Joshi [22] investigate WSD by leveraging external knowledge sources to resolve linguistic ambiguities. Their study aimed at developing a WSD tool for Hindi using Hindi WordNet and the LESK algorithm, achieves a notable accuracy rate of 71.4%. Mosima et.al [35] details a WSD system employing a knowledge-based method specifically designed for Sesotho sa Leboa, a South African language. This system’s performance surpasses the standard threshold, making it suitable for morphologically complex and resource-scarce languages. Nonetheless, it does not account for word morphology and character structures, which could limit its effectiveness with morphologically diverse terms.

O. Sainz et.al[37] investigates a knowledge-based approach to word sense disambiguation (WSD) applied to resources like BabelDomain, CSI and WordNet. It reveals strong links between specific domains and word senses, demonstrating effective domain-specific performance. However, the approach is limited to a narrow range of domains, restricting its broader application across different contexts. Basile et al. [7] present a novel WSD algorithm that builds upon the Lesk method by integrating a word similarity function based on distributional semantics. Leveraging BabelNet's extensive sense inventory, the algorithm outperforms standard benchmarks and competitors in SemEval-2013 for English. This method represents glosses and context as vectors in a semantic space, showing promising results, especially when using sense frequencies. Future work may involve adapting the algorithm to specific domains and expanding its applicability to other languages.

Arindam Roy et al. [36] achieved an accuracy of approximately 75% in their exploration of Nepali language processing, employing a comprehensive framework that combines overlap-based, conceptual distance-based and semantic-based methods.

Eniafe et al. [2] develop an optimized WSD system using datasets from Wikipedia and the Semcor Corpus. Their system evaluates effectiveness category-wise within the English language domain. Alok Ranjan Pal et al. [8] introduce a hybrid WSD model for English, combining Modified Lesk and Bag-of-Words techniques. The model recursively disambiguates unmatched words, achieving exceptional performance. Prity Bala [15] examines WSD within the realm of NLP and artificial intelligence, emphasizing its theoretical and practical importance. The study focuses on Hindi, using a knowledge-based approach with Hindi WordNet and POS tagging to disambiguate words, offering insights into challenges related to POS tagging accuracy.

Chandra Bal Singh Gautum et al. [13] apply the Lesk algorithm to disambiguate bigrams and trigrams in Hindi sentences with polysemous verbs. Utilizing Hindi WordNet and manual evaluation, the approach achieves an average precision of 52.98% for bigrams and 37.04% for trigrams. The study highlights challenges in handling polysemy in Hindi and assesses the Lesk algorithm's effectiveness in this context, revealing its strengths and areas for improvement. Jagbir Singh and Iqbal Singh [5] conducted a Punjabi WSD study, collecting instance and accuracy data while experimenting with various context window sizes. Their findings highlight the relationship between context window size and WSD accuracy, utilizing the Enhanced Lesk Algorithm and Indo-Wordnet data. Sarika et al. [19] explore WSD in Hindi using the Lesk algorithm, focusing on ambiguous verbs from Hindi WordNet. The study achieves an average precision of 52.98% for bigram words and 37.04% for trigram words and an average recall of 33.17% for bigram words and 18.56% for trigrams, demonstrating the Lesk algorithm's effectiveness in resolving polysemy in Hindi.

Andrei Minca and Ştefan Diaconescu [1] introduce a WSD technique that focuses on semantic similarity among word senses, utilizing semantic trees and a phrase-based window for context analysis, incorporating the latest WordNet glosses. Mohannad AlMousaa et al. [6] present the SCSMM algorithm, a new WSD approach integrating semantic similarity, heuristic knowledge and document context to address word sense ambiguity. The algorithm preserves word order while maximizing sentence context, demonstrating high performance in noun disambiguation and suggesting future research directions in topic modeling and semantic similarity measures.

AKM Sabbir et al. [8] explore biological WSD using the publicly available MSH WSD dataset. Their study combines knowledge-based methods with recent advancements in neural embeddings. By using MetaMap for concept mapping, their model achieves an impressive accuracy of 92.24% and proposes a more advanced nearest neighbor approach. J. Degraeuwe et.al [38] achieves an impressive F1 score of 0.8836, addressing the absence of "word sense awareness" features in existing WSD systems. This high score indicates the system's accuracy in determining correct word senses. However, its application is limited by its focus on a few domains, potentially reducing its effectiveness across varied linguistic scenarios.

Various methodologies have been adopted and studied for WSD in numerous languages. Data play a critical role in choosing the methodology. Factors like size of the data, annotation, nature of the language etc. were closely lookup for the suitability of the methodology. Hence, for the Manipuri language, the knowledge based approach will prove the best seeing the size, type and nature of the language.

3. Proposed methodology

3.1 Preprocessing

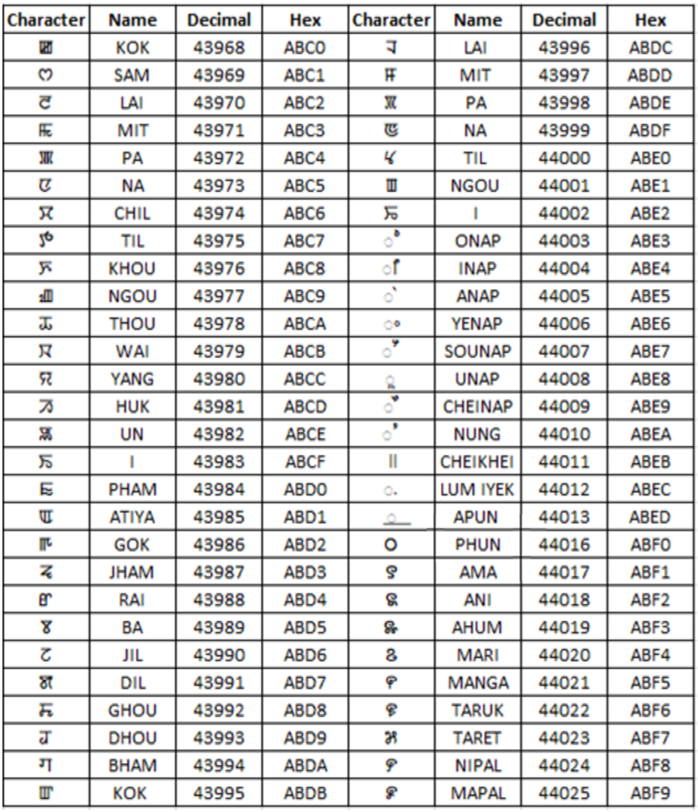
All the raw data collected are mostly in unstructured form and are not useful and may contains unwanted data. Preprocessing is the process that involve the conversion of unstructured data into the structured data. The collected data for this research work are also not in the format which are understable by the machine[34]. The detailed description of the collected data are given in the below table.

Table 1: Detailed description of the collected data

Name of the resource	Script	Format	Remark
Manipuri Synset (IndoWordNet data)	Meitei Mayek	Non – Unicode	Required in Unicode format
Manipuri Corpus (Monolingual corpus)	Bengali	Unicode	Need in Meitei Mayek Unicode format
Electronic Dictionary	Bengali	Unicode	Need in Meitei Mayek Unicode format

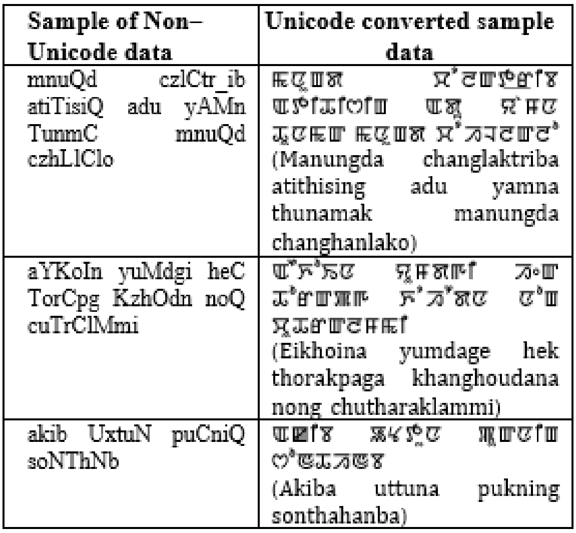
Thus, these non-unicode data has to be converted into the Unicode format. The conversion of the data into the proper Meetei/Meitei mayek Unicode format is done programtically by using the Meetei/Meitei mayek Unicode character set. The Meitei/Meetei mayek scripts unicode values are used to mapped with every corresponding non-unicode character[34]. The Meitei Mayek Unicode character set is given in the below table.

Table 2: Meitei/Meetei mayek unicode character values



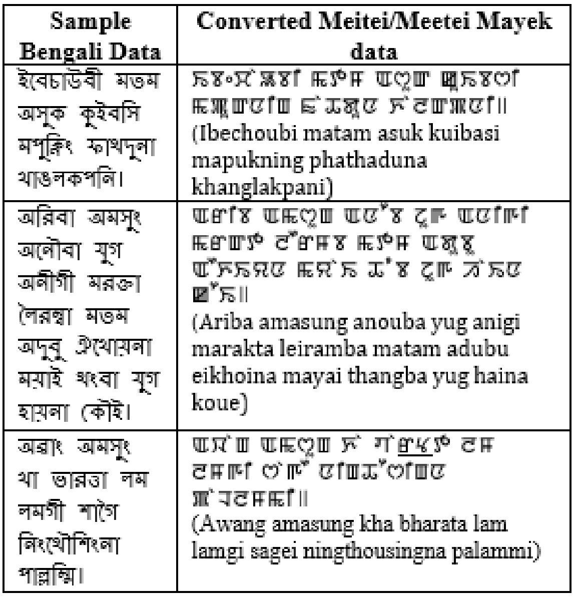
Character	Name	Decimal	Hex	Character	Name	Decimal	Hex
ꯀ	KOK	43968	ABC0	ꯜ	LAI	43996	ABDC
ꯁ	SAM	43969	ABC1	ꯝ	MIT	43997	ABDD
ꯂ	LAI	43970	ABC2	ꯞ	PA	43998	ABDE
ꯃ	MIT	43971	ABC3	ꯟ	NA	43999	ABDF
ꯄ	PA	43972	ABC4	ꯠ	TIL	44000	ABE0
ꯅ	NA	43973	ABC5	ꯡ	NGOU	44001	ABE1
ꯆ	CHIL	43974	ABC6	ꯢ	I	44002	ABE2
ꯇ	TIL	43975	ABC7	◌ꯣ	ONAP	44003	ABE3
ꯈ	KHOU	43976	ABC8	◌ꯤ	INAP	44004	ABE4
ꯉ	NGOU	43977	ABC9	◌ꯥ	ANAP	44005	ABE5
ꯊ	THOU	43978	ABCA	◌ꯦ	YENAP	44006	ABE6
ꯋ	WAI	43979	ABCB	◌ꯧ	SOUNAP	44007	ABE7
ꯌ	YANG	43980	ABCC	◌ꯨ	UNAP	44008	ABE8
ꯍ	HUK	43981	ABCD	◌ꯩ	CHEINAP	44009	ABE9
ꯎ	UN	43982	ABCE	◌ꯪ	NUNG	44010	ABEA
ꯏ	I	43983	ABCF	꯫	CHEIKHEI	44011	ABEB
ꯐ	PHAM	43984	ABD0	◌꯬	LUM IYEK	44012	ABEC
ꯑ	ATIYA	43985	ABD1	◌꯭	APUN	44013	ABED
ꯒ	GOK	43986	ABD2	꯰	PHUN	44016	ABF0
ꯓ	JHAM	43987	ABD3	꯱	AMA	44017	ABF1
ꯔ	RAI	43988	ABD4	꯲	ANI	44018	ABF2
ꯕ	BA	43989	ABD5	꯳	AHUM	44019	ABF3
ꯖ	JIL	43990	ABD6	꯴	MARI	44020	ABF4
ꯗ	DIL	43991	ABD7	꯵	MANGA	44021	ABF5
ꯘ	GHOU	43992	ABD8	꯶	TARUK	44022	ABF6
ꯙ	DHOU	43993	ABD9	꯷	TARET	44023	ABF7
ꯚ	BHAM	43994	ABDA	꯸	NIPAL	44024	ABF8
ꯛ	KOK	43995	ABDB	꯹	MAPAL	44025	ABF9

Table 3: Sample 1 of unicode converted data



Sample of Non–Unicode data	Unicode converted sample data
mnuQd czlCtr_ib atiTisiQ adu yAMn TunmC mnuQd czhLlClo	ꯃꯅꯨꯡꯗ ꯆꯪꯂꯛꯇ꯭ꯔꯤꯕ ꯑꯇꯤꯊꯤꯁꯤꯡ ꯑꯗꯨ ꯌꯥꯝꯅ ꯊꯨꯅꯃꯛ ꯃꯅꯨꯡꯗ ꯆꯪꯍꯜꯂꯛꯂꯣ (Manungda changlaktriba atithising adu yamna thunamak manungda changhanlako)
aYKoIn yuMdgi heC TorCpg KzhOdn noQ cuTrClMmi	ꯑꯩꯈꯣꯏꯅ ꯌꯨꯝꯗꯒꯤ ꯍꯦꯛ ꯊꯣꯔꯛꯄꯒ ꯈꯪꯍꯧꯗꯅ ꯅꯣꯡ ꯆꯨꯊꯔꯛꯂꯝꯃꯤ (Eikhoina yumdage hek thorakpaga khanghoudana nong chutharaklammi)
akib UxtuN puCniQ soNThNb	ꯑꯀꯤꯕ ꯎꯠꯇꯨꯟ ꯄꯨꯛꯅꯤꯡ ꯁꯣꯟꯊꯍꯟꯕ (Akiba uttuna pukning sonthahanba)

Table 4: Sample 2 of unicode converted data



Sample Bengali Data	Converted Meitei/Meetei Mayek data
ইবেচাউবী মতম অসুক কুইবসি মপুক্নিং ফাথদুনা খাঙলকপনি।	ꯏꯕꯦꯆꯧꯕꯤ ꯃꯇꯝ ꯑꯁꯨꯛ ꯀꯨꯢꯕꯁꯤ ꯃꯄꯨꯛꯅꯤꯡ ꯐꯥꯊꯗꯨꯅꯥ ꯈꯥꯡꯂꯛꯄꯅꯤ॥ (Ibechoubi matam asuk kuibasi mapukning phathaduna khanglakpani)
অরিবা অমসুং অনৌবা যুগ অনীগী মরক্তা লৈরম্বা মতম অদুবু ঐখোয়না ময়াই থংবা যুগ হায়না কৌই।	ꯑꯔꯤꯕ ꯑꯃꯁꯨꯡ ꯑꯅꯧꯕ ꯌꯨꯒ ꯑꯅꯤꯒꯤ ꯃꯔꯛꯇ ꯂꯩꯔꯝꯕ ꯃꯇꯝ ꯑꯗꯨꯕꯨ ꯑꯩꯈꯣꯢꯅ ꯃꯌꯥꯢ ꯊꯡꯕ ꯌꯨꯒ ꯍꯥꯢꯅ ꯀꯧꯏ॥ (Ariba amasung anouba yug anigi marakta leiramba matam adubu eikhoina mayai thangba yug haina koue)
অৱাং অমসুং খা ভারতা লম লমগী শাগৈ নিংথৌশিংনা পাল্লম্মি।	ꯑꯋꯥꯡ ꯑꯃꯁꯨꯡ ꯈꯥ ꯚꯥꯔꯇ ꯂꯝ ꯂꯝꯒꯤ ꯁꯥꯒꯩ ꯅꯤꯡꯊꯧꯁꯤꯡꯅ ꯄꯥꯜꯂꯝꯃꯤ॥ (Awang amasung kha bharata lam lamgi sagei ningthousingna palammi)

After the data are converted into Unicode format, they are further checked for the spelling mistakes. To reduce the processing time, various unnecessary and unimportant words commonly not as stop words are removed[33, 34]. Some cases also perform stemming[26].

3.2 Prerequisites of implementation of Lesk algorithm

3.2.1 Extraction of ambiguous words

Ambiguous words are those that can lead to multiple interpretations or lack precise meaning. A term is considered ambiguous under the following conditions:

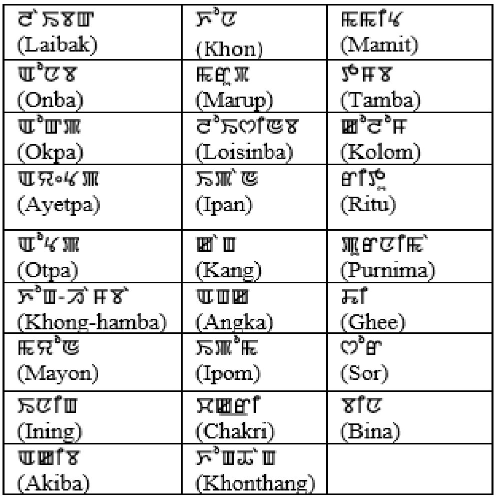
a) When it includes multiple synonyms in the IndoWordNet.

b) When it is used in different parts of speech.

c) When it has several meanings in a dictionary.

In this research paper, we have selected a specific set of 26 ambiguous Manipuri words, considering various factors relevant to the challenges of Word Sense Disambiguation (WSD) tasks. Manipuri is a tonal language, but most available electronic data do not use “lum” in writing tonal words. Despite debates over the use of “lum” in the Meitei/Meetei mayek script, selective ambiguous sentences are considered for testing purpose.

Table 5: Sample list of Manipuri Ambiguous words



ꯂꯥꯏꯕꯛ (Laibak)	ꯈꯣꯟ (Khon)	ꯃꯃꯤꯠ (Mamit)
ꯑꯣꯟꯕ (Onba)	ꯃꯔꯨꯞ (Marup)	ꯇꯝꯕ (Tamba)
ꯑꯣꯛꯄ (Okpa)	ꯂꯣꯏꯁꯤꯟꯕ (Loisinba)	ꯀꯣꯂꯣꯝ (Kolom)
ꯑꯌꯦꯠꯄ (Ayetpa)	ꯏꯄꯥꯟ (Ipan)	ꯔꯤꯇꯨ (Ritu)
ꯑꯣꯠꯄ (Otpa)	ꯀꯥꯡ (Kang)	ꯄꯨꯔꯅꯤꯃꯥ (Purnima)
ꯈꯣꯡ-ꯍꯝꯕꯥ (Khong-hamba)	ꯑꯡꯀ (Angka)	ꯘꯤ (Ghee)
ꯃꯌꯣꯟ (Mayon)	ꯏꯄꯣꯝ (Ipom)	ꯁꯣꯔ (Sor)
ꯏꯅꯤꯡ (Ining)	ꯆꯀꯔꯤ (Chakri)	ꯕꯤꯅ (Bina)
ꯑꯀꯤꯕ (Akiba)	ꯈꯣꯟꯊꯥꯡ (Khonthang)	

3.2.2 Sense inventory

A sense inventory serves as a lexical repository that systematically categorizes and organizes words according to their distinct meanings or senses. This structured framework enables a comprehensive representation of the various interpretations linked to individual words, thereby enhancing the understanding of language complexity. For the Manipuri language, the sense inventory encompasses the meanings of words derived from sources such as IndoWordNet and the Manipuri dictionary[29]. Additionally, linguists have manually encoded and incorporated supplementary words into the sense inventory, ensuring a thorough and detailed lexical resource.

3.2.3 Implementation of Lesk algorithm

The Lesk algorithm is a pioneering method in the field of natural language processing for resolving word sense ambiguity. Proposed by Michael Lesk in 1986, this algorithm aims to determine the most appropriate sense of a polysemous word by analyzing its context and comparing it to predefined senses in a dictionary or a thesaurus[16].

The core idea of the Lesk algorithm involves matching the context of an ambiguous word in a text with the definitions of possible senses of the word. To achieve this, the algorithm follows a series of steps:

I) Context Extraction: It starts by identifying the surrounding words or phrases of the ambiguous term in the given text. This context is crucial as it provides clues about the intended meaning of the word.

II) Sense Definitions Retrieval: Next, the algorithm retrieves the definitions for each possible sense of the ambiguous word from a lexical database. These definitions serve as reference points for comparison.

III) Overlap Calculation: The algorithm then computes the overlap between the words in the context and the words in each sense's definition. This involves comparing the contextual words with the words present in the definitions of the possible senses.

IV) Sense Selection: The sense with the highest overlap—the most words in common between the context and the definition—is chosen as the most likely meaning of the ambiguous word.

Let us consider the following sentences in Manipuri language:

ꯀꯩ ꯑꯁꯤ ꯁꯥ ꯃꯈꯜ ꯑꯃꯅꯤ ॥

(Tiger is a kind of animal.)

ꯀꯩ ꯗ ꯆꯦꯡ ꯊꯝꯂꯩ ॥

(Rice is usually stored in granary.)

In the given sentences, the first sentence pertains to the sense related to animals whereas the second pertains to the granary context, as indicated by the words "animal" and "rice" which align with the respective senses of the terms. To determine the correct meaning, an algorithmic approach utilizes definitions and Part-Of-Speech (POS) tags for comparison.

Algorithm:

Disambiguation Based on Lesk Algorithm

```
Input:
  Ambiguous word
Output:
  Index of the highest weight of the overlapped sense
1. For each sense (Sense_of_Target_Word_i) in Senses_of_Target_Word, do
2.   Initialize Counter_i to 0
3.   For each word (Word_j) in Context_Words_in_Document except the target word, do
4.     Initialize Max_Score_j to 0
5.     For each sense (Sense_of_Word_k) in Senses_of_Word(Word_j), do
6. If Max_Score_j < Calculate_Relatedness (Sense_of_Target_Word_i, Sense_of_Word_k), then
7.       Update Max_Score_j with Calculate_Relatedness(Sense_of_Target_Word_i,
         Sense_of_Word_k)
8.       If Max_Score_j > Threshold, then
9.         Increment Counter_i by Max_Score_j
10. Return Index such that Counter_i ≥ Counter_j for all j, where j ranges from 1 to n and n is the number of words in the sentence.
```

The detailed functioning of the Lesk algorithm, specifically LESK_Ori, is illustrated below:

E.g: "ꯃꯍꯥꯛꯀꯤ ꯅꯨꯡꯉꯥꯏꯇꯕ ꯋꯥꯔꯤ ꯇꯔꯥꯒꯥ ꯑꯩꯒꯤ ꯃꯤꯠꯀꯤ

ꯄꯤ ꯁꯤꯟꯊꯔꯥꯛꯂꯝꯃꯤ॥"

(Mahakki nungaitaba wari taraga eige mitage pi sintharaklammi)

Here, the underlined word has two meanings. The two meanings are tear and to give or sacrifice. The glosses of these two words in IndoWordNet are shown below:

Sense 1(tear): ꯑꯋꯥꯕ, ꯅꯨꯡꯉꯥꯏꯇꯕ ꯅꯠꯇꯔꯒꯥ ꯀꯍꯦꯟꯅ

ꯅꯨꯡꯉꯥꯏꯇꯕ ꯃꯇꯝꯗ ꯃꯤꯠꯀꯤ ꯇꯥꯔꯛꯄ ꯄꯤꯊꯣꯛꯐꯝꯗꯒꯤ

ꯊꯣꯔꯛꯄ ꯑꯌꯥꯛꯄ ꯄꯣꯠ

(Awaba, nungaitaba natraga kahenna nungaitaba matamda mittage tarakpa pithokphamdage thorakpa ayakpa pot)

Sense 2(to give/sacrifice): ꯀꯔꯤꯒꯨꯝꯕ ꯑꯈꯟꯅꯕ ꯊꯕꯛ,
ꯃꯤꯁꯛ ꯅꯠꯇꯔꯒꯥ ꯃꯔꯝꯅꯆꯤꯡꯕꯒꯤ ꯗꯥꯃꯛ ꯂꯩꯖꯕ ꯄꯨꯝꯕ ꯄꯤꯕ (Karigumba akhannaba thabak, misak nattraga maramnachingbage damak leijaba pumba piba)

In this instance, the first sense is identified as the correct interpretation because it contains two words that match those in the context, whereas the second sense does not have any overlapping words.

Certain modifications can also made to the Lesk algorithm that can handle other aspects of the word senses like those of the Adaptive Lesk algorithm[4]. Such techniques enhance the performance of the WSD. Hence, certain additional factors can be considered to improve the performance of the WSD.

3.3 Additional techniques for enhancing the baseline algorithm, the Lesk algorithm

To the Lesk algorithm, various factors can be additionally considered to cover up numerous loopholes in this algorithm. Variants of the Lesk algorithm can be obtained using the following methods:

3.3.1 LESK_Word_Rel

There are also the cases, where the meanings are disambiguated using the synonymous word of the ambiguous word[23]. Furthermore, besides synonym other relationships like hypernymy, hyponymy, meronymy, etc. associated with the ambiguous words also sometimes help in disambiguating the process. Forexample, consider the word "ꯃꯤ", the actual meaning

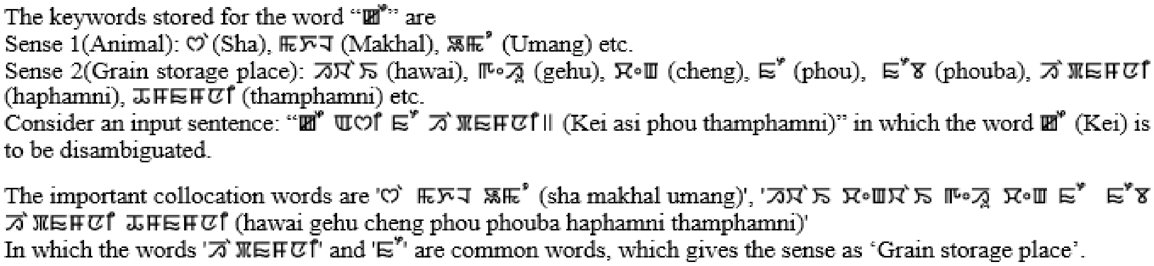
is found in the synonymous word "ꯃꯤꯑꯣꯏ". Hence, we

We include the synonym, hypernym, hyponym and meronym of the ambiguous words. Similar process was followed but with respect to these relations of the ambiguous word. Inclusion of these relations expand the word network and minutely catches the words that has been included in the gloss of the ambiguous word.

3.3.2 LESK_Key_Colloc

In knowledge based approach, best accuracy is achieved when accurate and maximum overlap occurs. To make most appropriate match, we need to bring those words which are very frequently go along with the ambiguous word. These frequent words are termed as keywords in this research work. Such frequent words can be found in the collocation words of ambiguous words. Thus, collocation refers to those words that go with a particular word. Class-based collocations play a crucial role in WSD [20]. For this research work, we consider five window words depending on the position of the words. These collocation words are found using the Manipuri corpus. For instance, if the particular word is a noun, we consider the significant words that follow the words and if the word is a verb then the significant words will be those words that are before the particular words. If the particular word is an adjective or an adverb then the words that are before and after will be considered.

For instance:



The keywords stored for the word “ꯀꯩ” are
Sense 1(Animal): ꯁꯥ (Sha), ꯃꯈꯜ (Makhal), ꯎꯃꯪ (Umang) etc.
Sense 2(Grain storage place): ꯍꯋꯥꯏ (hawai), ꯒꯦꯍꯨ (gehu), ꯆꯦꯡ (cheng), ꯐꯧ (phou), ꯐꯧꯕ (phouba), ꯍꯥꯞꯐꯝꯅꯤ (haphamni), ꯊꯝꯐꯝꯅꯤ (thamphamni) etc.
Consider an input sentence: “ꯀꯩ ꯑꯁꯤ ꯐꯧ ꯍꯥꯞꯐꯝꯅꯤ꯫ (Kei asi phou thamphamni)” in which the word ꯀꯩ (Kei) is to be disambiguated.

The important collocation words are 'ꯁꯥ ꯃꯈꯜ ꯎꯃꯪ (sha makhal umang)', 'ꯍꯋꯥꯏ ꯆꯦꯡꯍꯋꯥꯏ ꯒꯦꯍꯨ ꯆꯦꯡ ꯐꯧ ꯐꯧꯕ ꯍꯥꯞꯐꯝꯅꯤ ꯊꯝꯐꯝꯅꯤ (hawai gehu cheng phou phouba haphamni thamphamni)'
In which the words 'ꯍꯥꯞꯐꯝꯅꯤ' and 'ꯐꯧ' are common words, which gives the sense as ‘Grain storage place’.

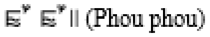
3.3.3 LESK_HCR

Some hand coded grammatical rules of the languages can be applied to disambiguate the ambiguous word[28]. These rules are basically useful for the duplication or repetitive words. Two important rules of the Manipuri language are considered:

II. Usually last word can be treated as ‘verb’ as the verb comes at last in a sentence in the Manipuri language.

This rule helps us to resolve mainly ambiguous repetitive words. When the ambiguous word is used repetitively, the last one will always be verb and the other will be of other part-of-speech(POS). Thus, because of this different POS, differentiation of the sense(s) becomes narrower and disambiguate easily[24].

For illustration purpose, let us consider the sentence:



ꯐꯧ ꯐꯧ꯫ (Phou phou)

In the above sentence, the word “ꯐꯧ” refers to noun as well as verb. The first “ꯐꯧ” is noun while the second “ꯐꯧ” is verb.

In the noun sense, the word “ꯐꯧ” means food grain(rice) while in the verb sense, it refers to the act of drying something.

Thus, using the above algorithm which inputs the ambiguous word along with its POS we can disambiguate the word by matching the input POS and the matching sense’s POS of the ambiguous word.

III. For those words which exhibit multiple POS, refer to only those POS which has been given along with the input sentence:

Consider the sentence in which the word "ꯑꯣꯟꯅꯕ" needs to be disambiguated with respect to noun POS:

ꯑꯌꯨꯛ ꯅꯨꯡꯊꯤꯜꯒꯤ ꯑꯣꯟꯅꯇꯩꯅꯕ ꯋꯥꯍꯩꯗꯦ ꯅꯨꯡꯃꯤꯗꯥꯡꯅꯤ॥ (Ayuk nungthilgi onnateinaba waheide nungmidangni)

The word "ꯑꯣꯟꯅꯇꯩꯅꯕ" refers to noun as well as adjective. In the above sentence, it refers to noun sense.
The possible senses of the word are:
Sense 1(Opposite): ꯋꯥꯍꯩ ꯑꯃꯒꯤ ꯋꯥꯈꯜꯗꯒꯤ ꯃꯗꯨꯒꯤ ꯋꯥꯍꯟꯊꯣꯛꯀꯤ ꯇꯣꯡꯥꯟꯕ ꯋꯥꯍꯟꯊꯣꯛ ꯄꯤꯕ ꯑꯇꯣꯞꯄ ꯋꯥꯍꯩ (Wahei amage wakhaldage madugi wahanthokki tonganba wahanthok piba attoppa wahei)
Sense 2(Wrong): ꯑꯆꯨꯝꯕ ꯑꯣꯢꯗꯕ (Achumba oidaba)
Sense 3(Reverse): ꯃꯇꯨꯡ ꯍꯟꯕ (Matung hanba)
Sense 4(False): ꯐꯠꯇꯕ ꯁꯤꯖꯤꯟꯅꯕ ꯌꯥꯗꯕ ꯑꯣꯢꯕ (Fattaba sijinnaba yadaba oiba)

The first and the third senses are of noun while the second and the fourth are of adjective POS. In noun senses, the first sense means opposite while the third sense refers to reverse. In adjective senses, the second sense refers to wrong while the fourth sense refers to false.

As we are interested in disambiguating the noun sense, only first and the third senses will be considered. Here, the first sense will be the winner as there are two overlap words while the other did not have any matched word.

The main benefits of using this technique is that the divide and conquer rule can be applied to a word with the capability of exhibiting different POS as we can ignore the unmatched POS of the ambiguous words thereby reducing the searching and computational time.

This rule is to handle mainly repetitive words. Some of the ambiguous words have the capability of exhibiting different POS. Thus, candidate senses of the ambiguous words can be restricted to only the matched POS with that of the inputted POS of the ambiguous word.

For instance, consider the word "ꯏ (E)".

The different senses of ꯏ from the Sense Inventory are:

Sense 1(Noun): ꯊꯋꯥꯏ ꯄꯥꯟꯕꯁꯤꯡꯒꯤ ꯁꯤꯡꯂꯤꯗ ꯆꯦꯟꯂꯤꯕ ꯃꯍꯤ ꯂꯥꯡꯕ ꯑꯉꯥꯡꯕ ꯄꯣꯠ

(Thawai panbasingge singlida chenliba mahi langba angangba pot)

Sense 2(Noun): ꯏꯅ ꯀꯨꯞꯄ ꯌꯨꯝꯒꯤ ꯃꯊꯛ

Sense 3(Verb): ꯃꯌꯦꯛꯇ ꯏꯗꯨꯅ ꯊꯝꯕ (Ena kuppa yumge mathak)

(Mayekta eduna thamba)

If the inputted sentence is "ꯅꯣꯡꯖꯨꯊꯗ ꯏ ꯒꯦ ꯌꯨꯝꯊꯛꯇꯒꯦ ꯏꯁꯤꯡ ꯇꯥꯔꯀꯦ॥

(Nongjuthada e ge yumthaktage esing tarake)".

The word to be disambiguated is "ꯏ" and the POS of the word is Verb. Then, instead of further applying overlapping method we can directly conclude that the correct sense is of sense 3.

3.3.4 Lesk_Key_Unco

This technique is used only after an overlapping algorithm is applied to the inputted ambiguous sentence. Many a time it happens that a word may be used with a very unusual word or used along with a very least used word. Such existence may reduce the performance of the system because the above algorithm fails to comply with such a situation. Hence to include all the uncovered collocated words of the Manipuri ambiguous words, we reused the unmatched words from the inputted sentence. These unmatched words are further analyzed and only the significant words are entered in the Bag-of-words of those ambiguous words. The unmatched significant words are considered for future disambiguous reference as these unmatched words imply that such words can be used with the ambiguous word to reflect a meaning. This method indirectly improve the performance by increasing the size and the content of the database.

For instance,

ꯄꯣꯠ ꯄꯨꯊꯣꯛꯂꯤꯕ ꯃꯤ ꯑꯃꯅ ꯐꯦꯛꯇꯣꯔꯤꯒꯤ ꯃꯃꯂꯒꯤ
ꯃꯇꯦꯡꯅ ꯄꯣꯠ ꯄꯨꯊꯣꯛꯄꯒꯤ ꯑꯆꯥꯡꯃꯜ, ꯇꯣꯡꯉꯥꯟ ꯇꯣꯡꯉꯥꯟꯕ ꯄꯣꯊꯣꯛꯀꯤ ꯅꯤꯌꯥꯝꯒꯤ ꯃꯇꯨꯡ ꯏꯟꯅ ꯂꯦꯞꯄ ꯉꯝꯃꯦ॥

(Pot puthokliba mi amana factoryge mamalge matengna pot puthokpage achangmal, tongan tonganba pothokki niyamge matung enna leppa ngame)

Significant words which are not in the glosses of the ꯃꯤ are collected and added to the keyword database. The significant words selected are ꯐꯦꯛꯇꯣꯔꯤꯒꯤ (factoryge), ꯃꯃꯂꯒꯤ (mamalge), ꯅꯤꯌꯥꯝꯒꯤ (niyamge) and ꯑꯆꯥꯡꯃꯜ(Achangmal). These words are added into the keywords of the "ꯃꯤ" human senses. This updated database when help in disambiguating the uncatchable word sense.

4. Results and discussion

Our study aimed to improve word sense disambiguation in the Manipuri language, a language with complex polysemy and context-dependent meanings. All the proposed methods are tested on 1170 sentences which contains ambiguous words. Using this model the original Lesk algorithm can correctly disambiguate 42.56%. The Ad-on factors that we supplement to the existing algorithm provide a significant improvement. Various loopholes are covered, most impactfully by the Lesk_Key_Colloc. The Lesk_Key_Colloc Various reasons to use the above factors are listed below:

Table 6: Methods and their usage

Methods	Usage
Lesk_Rel	Catch up those collection of words which have the same sense like those synonymous word. Futher, it can also find out those senses which can be indirectly referred by words relationship like antonym, hypernymy, hyponymy etc.
Lesk_Key_Colloc	Those frequently go words that are not in the sense inventory but are commonly used in daily communication. This method enhances the overlapping percentages of words that can converge to the derivation of the correct sense.
Lesk_HCR	Commonly able to handle the correct sense of the reduplicated words which are very difficult to handle by the machine. Can also easily identify the correct sense if the

	ambiguous word exist with different part-of-speech. Comprehensively improve the system's computational time.
Lesk_Key_Unco	This method is useful in those cases where very rarely used words or ancient words can be included in the data repository. Mostly helpful in increasing the size of the data repository and the completeness of data content of all spheres. This method is used to complement the other above methods.

Table 7: System performance using the Ad-on techniques

Methods	Improvement in correctly disambiguated percentage
Lesk_Ori	42.56
Lesk_Ori & Lesk_Rel	54.01
Lesk_Ori, Lesk_Rel & Lesk_Key_Colloc	76.75

Initially, the model is tested with the Lesk algorithm. the result is tested on 1170 sentences which consist of 59 ambiguous words. The results show a significant increase in disambiguation accuracy. One key innovation of our approach is the integration of contextual information. By incorporating not only the target word but also the nearby words and the whole sentence context, we achieved a substantial reduction in ambiguous interpretations.

The accuracy of this approach using various methods ranges from 42.56% to 76.75%. To the original Lesk algorithm, which achieved the lowest accuracy of 42.56%, we added word relations and word collocations in an incremental manner to achieved a promising accuracy of 76.75%. Special cases are handled using hand coded rules for repetitive words and uncovered keywords for very unusual and rarely used words. Considering the limited work carried out for WSD system in Manipuri language, this approach performs quite well. Since the senses of the words are annotated by the human expert, for some words there are cases where the different combinations of words are used to mean a single word or vice – versa. If the numbers of sense annotated Manipuri words in IndoWordNet is increased , this WSD approach will work wonderfully.

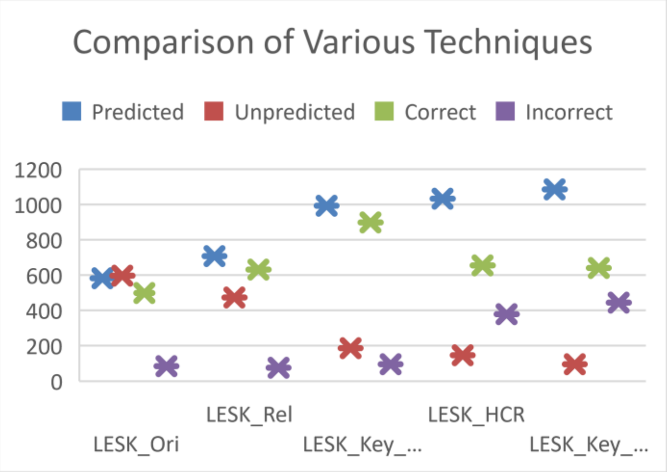


Fig 1: Comparative vision of various methodologies

The figure above presents a comparative analysis of experimental data across different methodologies. It displays both the anticipated and unanticipated outcomes, as well as the accuracy of the results and also infers us that every method is applicable to solve a limited problem of WSD.

The outcomes from the Manipuri Word Sense Disambiguation (WSD) experiments highlight differences in effectiveness among various disambiguation techniques. From these results, we can derive several key observations:

I) Method Efficiency: The LESK_Key_Colloc and LESK_Key_Unco methods achieved the highest accuracy in predicting the correct sense of words, with LESK_HCR also performing well. These techniques are notably efficient for disambiguating word senses within Manipuri text.

II) Areas for Enhancement: While LESK_Rel made a considerable number of correct predictions, it also yielded a significant amount of incorrect ones. This indicates that there is potential for improving the precision of this method.

III) Unresolved Cases: Certain methods, such as LESK_Ori and LESK_Rel, left a significant number of sentences without predictions. Investigating these cases further is essential to understand the underlying issues and improve these methods.

IV) Role of Specialized Rules: The LESK_HCR method, which incorporates domain-specific rules, achieved a high success rate. This highlights the value of integrating specialized knowledge and linguistic insights into WSD approaches.

5. Conclusion and future work

In a world that values effective cross-cultural communication, developing Word Sense Disambiguation (WSD) techniques for languages such as Manipuri presents both a technical challenge and a cultural necessity. This study has made significant strides in enhancing Word Sense Disambiguation (WSD) for the Manipuri language by developing a novel system that integrates various techniques with the Lesk algorithm. Addressing the unique challenge of ambiguous words with identical spellings, the proposed method emphasizes contextual analysis, moving beyond traditional reliance on lemmatization. By incorporating a comprehensive Sense Inventory, IndoWordNet relationship data and Dictionary concepts, the system offers notable improvements over existing WSD approaches for Manipuri.

The integration of word relations, collocations, hand-coded rules and keyword identification has led to a marked increase in disambiguation accuracy, with performance metrics ranging from 42.56% to 76.75%. These enhancements demonstrate that our approach not only refines the Lesk algorithm but also introduces a level of precision that surpasses current systems. Through rigorous testing on 1,170 sentences containing ambiguous words, the developed system has proven its efficacy in tackling the complexities of Manipuri language processing.

This advancement contributes substantially to the field of Natural Language Processing (NLP), providing a robust framework for improving computational understanding of the Manipuri language. The methods and findings presented in this research offer valuable insights and set a new standard for future developments in WSD for low-resource languages, highlighting the potential for similar approaches to benefit other linguistic contexts.

Some limitations include difficulties with overlapping words not precisely matched in the Sense Inventory, despite containing similar meanings through different word combinations. The small size of Manipuri data restricts the scope of research and manually coded rules can only cover so many possibilities. Additionally, the use of a 7-word window for collocations means some significant words may be omitted. To address these issues, we can adopted a hybrid approach that integrates various methods into a single model.

Acknowledgment

None.

Conflicts of interest

The authors have no conflicts of interest to declare.

References

1. A. Minca and S. Diaconescu. An approach to knowledge-based Word Sense Disambiguation using semantic trees built on a WordNet lexicon network. In proceedings of the 6th Conference on Speech Technology and Human-Computer Dialogue (SpeD). 2011 (pp. 1-6) doi: 10.1109/SPED.2011.5940744.
2. Eniafe Festus Ayetiran, Kehinde Agbele. An optimized Lesk-based algorithm for word sense disambiguation. De Guyter Open Computer Science 2018; 8:165-172.
3. Alok Ranjan Pal, Anirban Kundu, Abhay Singh, Raj Shekhar, Kunal Sinha. A hybrid approach to word sense disambiguation combining supervised and unsupervised learning. International Journal of Artificial Intelligence & Applications. 2013; 4(4):89- 101
4. Banerjee, S., Pedersen, T. An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet. Computational Linguistics and Intelligent Text Processing, Springer; 2002.p. 136-145.
5. Jagbir Singh, Iqbal Singh. Word Sense Disambiguation: Enhanced Lesk Approach in Punjabi Language, International Journal of Computer Applications . 2015; 129(6):23-27.
6. Mohannad AlMousaa, Rachid Benlamria, Richard Khoury. A Novel Word Sense Disambiguation Approach Using WordNet Knowledge Graph. Computer Speech and Language. 2022; 74. https://doi.org/10.1016/j.csl.2021.101337
7. Basile, Pierpaolo & Caputo, Annalina & Semeraro, Giovanni. An Enhanced Lesk Word Sense Disambiguation Algorithm through a Distributional Semantic Model. In proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, 2014 (pp. 1591–1600) Dublin City University and Association for Computational Linguistics.
8. AKM Sabbir, Antonio Jimeno-Yepes, Ramakanth Kavuluru. Knowledge-Based Biomedical Word Sense Disambiguation with Neural Concept Embeddings. In proceedings of the IEEE International Symposium Bioinformatics Bioeng. 2017 (pp. 163–170).
9. Bhattacharyya P., Fellbaum C., Vossen P. Principles, Construction and Application of Multlingual Wordnets, In proceedings of the 5th Global Wordnet Conference 2010. Narosa Publishing House.
10. Alok Chakrabarty, Bipul Syam Purkayastha, Lavya Gavshinde. Knowledge-Based Contextual Overlap keen Ideas for Word Sense Disambiguation using Wordnet. In proceedings of the 3rd Indowordnet workshop under the aegis of the 8th International Conference on Natural Language Processing. 2010

11. Basile, Pierpaolo & de Gemmis, Marco & Lops, Pasquale & Semeraro, Giovanni. Combining Knowledge-based Methods and Supervised Learning for Effective Italian Word Sense Disambiguation. In proceedings of the Semantics in Systems for Text Processing. 2008 (pp. 5-16) College Publications
12. Manish Kumar, Prasenjit Mukherji, Manik Hendre, Manish Godse, Baisakhi Chakraborty. Adaptive Lesk Algorithm Based Word Sense Disambiguation using the Context Information. International Journal of Advance Computer Science and Applications. 2020; 11(3):254-260.
13. Chandra Bhal Singh Gautam, Dilip Kumar Sharma. Hindi Word Sense Disambiguation Using Lesk Approach on Bigram and Trigram Words. In proceedings of the International Conference on Advances in Information Communication Technology & Computing. 2016 (pp. 1-5)
14. Ch. Yashwanta Singh. Manipuri Grammar, Rajesh Publications; 2000
15. Prity Bala. Knowledge Based Approach for Word Sense Disambiguation using Hindi Wordnet. The International Journal Of Engineering And Science (IJES). 2013. 2(4), 36-41
16. M. Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In proceedings of the 5th Annual International Conference on Systems Documentation. 1986 (pp. 24-26)Association for Computing Machinery
17. P. Boruah. A Novel Approach to Word Sense Disambiguation for a Low-Resource Morphologically Rich Language. In proceedings of the IEEE 6th Conference on Information and Communication Technology (CICT) 2022 (pp. 1-6) IEEE.
18. Inkpen, D. Z., G. Hirst. Automatic sense disambiguation of the near-synonyms in a dictionary entry. In proceedings of the 4th International Conference on Computational Linguistics and Intelligent Text Processing. 2003 (pp. 258–267) Springer, Berlin, Heidelberg
19. Sarika and Sharma, D. K.. Hindi Word Sense Disambiguation using Cosine Similarity. In proceedings of the International Conference on Information and Communication Technology for Sustainable Development (ICT4SD). 2015 (pp. 801-808) Springer
20. Tom O'Hara, Rebecca Bruce, Jeff Donner, Janyce Wiebe. Class-based Collocations for Word-Sense Disambiguation. In proceedings of SENSEVAL – 3 the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text 2004 (pp. 199-202) Association for Computational Linguistics.
21. Turney, P. D. Similarity of semantic relations. Computational Linguistics. 2006; 32(3):379-416
22. Pooja Sharma and Nisheeth Joshi. Knowledge-Based Method for Word Sense Disambiguation by Using Hindi WordNet, Engineering, Technology & Applied Science Research. 2019; 9(2):3985-3989.
23. Zouaghi A, Merhbene L, Zrigui M, Word sense disambiguation for Arabic language using the variants of the Lesk algorithm. In proceedings of the WORLDCOMP. 2011 (pp. 561-567)
24. Vasilescu, F., P. Langlais, G. Lapalme. Evaluating variants of the Lesk approach for disambiguating words. In proceedings of the Fourth International Conference on Language Resources and Evaluation, 2004 (pp. 633 – 636) European Language Resources Association (ELRA)
25. Indowordnet.https://www.cfilt.iitb.ac.in/indowordnet/home#currentStatistics. Accessed July 15 2024.
26. S. Poireiton Meitei, Bipul Syam Purkayastha, H. Mamata Devi. Development of Manipuri Stemmer: a Hybrid Approach, In proceedings of the International Symposium on Advanced Computing and Communication (lSACC). 2015. IEEE
27. Pushpak Bhattacharyya. IndoWordNet, In proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10). 2010 (pp. 3785- 3792) European Language Resources Association (ELRA).
28. Mukti Desai, Mrs. Kiran Bhowmick. Word Sense Disambiguation. International Journal of Engineering Science Invention. 2013; 2(10):01-04.

29. S. Poireiton Meitei. Electronic Dictionary(Manipuri to English) And Its Impact In The Development Of Manipuri Language. 2015. Unpublished Thesis
30. Ali Alkhatlana, Jugal Kalitab, Ahmed Alhaddad. Word Sense Disambiguation for Arabic Exploiting Arabic WordNet and Word Embedding. The 4th International Conference on Arabic Computational Linguistic. 2018 (pp.50-60).
31. Nemika Tyagi, Dr. Sudeshna Chakraborty, Jyotsna, Aditya Kumar, Nzanzu Katasohire Romeo. Word Sense Disambiguation Models Emerging Trends: A Comparative Analysis. Journal of Physics: Conference Series. 2022; 2161(1):1-8. DOI: 10.1088/1742-6596/2161/1/012035
32. Ahmed H. Aliwy and Hawraa A. Taher. Word Sense Disambiguation: Survey Study. Journal of Computer Science. 2019; 15(7):1004-1011. DOI: 10.3844/jcssp.2019.1004.1011
33. P. Ramya, B. Karthik. Word Sense Disambiguation Based Sentiment Classification Using Linear Kernel Learning Scheme. Intelligent Automation & Soft Computing. 2023; 36(2):2370-2391. DOI: 10.32604/iasc.2023.026291
34. Chingakham Ponykumar Singh, H. Mamata Devi. Efficient Data Preparation for Manipuri Language Processing: Preprocessing Strategies for Word Sense Disambiguation. 2023; 7(6):432-441. DOI: 10.46647/ijetms.2023.v07i06.062
35. M. A. Masethe, H. D. Masethe, S. O. Ojo and P. A. Owolawi, “Word Sense Disambiguation Pipeline Framework for Low Resourced Morphologically Rich Languages,” SSRN Electron. J., 2023, DOI: 10.2139/ssrn.4332896.
36. Arindam Roy, Sunita Sarkar, Bipul Syam Purkayashtha. Knowledge Based Approaches To Nepali Word Sense Disambiguation. International Journal of Natural Language Computing(IJNLC). 2014; 3(3):51-63. DOI:10.5121/ijnlc.2014.3305.
37. O. Sainz, O. L. de Lacalle, E. Agirre and G. Rigau. What do Language Models know about word senses? Zero-Shot WSD with Language Models and Domain Inventories. 2023; https://doi.org/10.48550/arXiv.2302.03353
38. J. Degraeuwe, P. Goethals. Interactive Word Sense Disambiguation in Foreign Language Learning. In Proceeding of the. 11th Work. Nat. Lang. Process. Comput. Lang. Learn. 2022; 190:46–54. Department of Translation, Interpreting and Communication. doi: 10.3384/ecp190005.