

Grammar-Driven Text Segmentation for Context Understanding of Myanmar Language

myo thida

mthida@uic.edu

University of Illinois at Chicago

Nu Wei Thet

Thein Kyaw LWIN

Batangas State University

Research Article

Keywords: Grammar-Based Text Segmentation, Burmese Script, Computational Linguistics, Language Technology for Southeast Asia, Low-resource languages, Rule-based NLP, Language-aware AI

Posted Date: January 23rd, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-8408602/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Grammar-Driven Text Segmentation for Context Understanding of Myanmar Language

Myo Thida^{1,2*}, NuWei Thet^{2,3†} and Thein Kyaw Lwin^{2,3†}

¹Department of Computer Science, University of Illinois,
Chicago, USA.

²Department of Research and Development, MMDT, Myanmar.

³ Data Science Department, Batangas State University, Philippines.

*Corresponding author(s). E-mail(s): mthida@uic.edu;

Contributing authors: nuwaitthet@gmail.com;

thein.kyaw.lwin.official@gmail.com;

†NW.Thet contributed to coding, and TK. Lwin contributed to testing.

Abstract

Text segmentation is a foundational process in NLP, but Myanmar presents a particularly difficult case due to its non-segmented, syllabic writing system and complex morphophonology. While syllable-level tokenization has seen notable progress, word- or meaningful phrase-level segmentation remains underexplored. Existing approaches largely rely on dictionary-based algorithms that ignore grammatical structure and contextual cues. This research proposes a new text segmentation method that integrates linguistic features such as conjunctions, post-particles, and whitespace usage, which carry implicit syntactic and semantic information. The proposed method produces stable and linguistically motivated segmentation across both formal and informal text styles, achieving an intrinsic boundary-level segmentation F1-score of 0.9 and a token-level segmentation F1-score of 0.8 across three different datasets. Extrinsic evaluation using named entity recognition (NER) as a downstream task demonstrates competitive performance across multiple entity types, with F1-scores of approximately 0.7 for DATE, NUM, LOC, PER, and TIME, even when applied without retraining the NER model. These results highlight the effectiveness and practical applicability of grammar-aware segmentation for Myanmar and underscore its potential for downstream NLP tasks in low-resource settings.

Keywords: Grammar-Based Text Segmentation, Burmese Script, Computational Linguistics, Language Technology for Southeast Asia, Low-resource languages, Rule-based NLP, Language-aware AI

1 Introduction

Text segmentation is a fundamental process in natural language processing (NLP) that converts continuous text into discrete units suitable for computational analysis. This step is relatively straightforward for languages that employ whitespace or explicit delimiters to separate words. However, in languages such as Myanmar (Burmese), text segmentation remains a challenging problem due to the absence of explicit word boundaries and the script’s intricate orthographic and morphological characteristics.

The Myanmar script consists of thirty-three consonants, each containing an inherent vowel sound /a/. These consonants combine with various diacritics and medials to modify vowel quality or introduce additional consonantal sounds. For example, the base consonant “က” (ka) forms the syllable “ကျ” (kya) when combined with the medial diacritic “ျ” (ya yit). Such graphemic combinations represent the basic phonological units in written Myanmar, where a single syllable often corresponds to a complex grapheme cluster rather than a linear sequence of characters. Consequently, conventional tokenization algorithms designed for alphabetic or space-separated languages are not directly applicable to Myanmar text.

In contrast to English, where words are composed of discrete characters from a finite alphabet and separated by whitespace, Myanmar words are written continuously, and certain consonants can appear either independently or in clusters. Some consonants, such as “က,” function as complete syllables, whereas others, including “င” (nga), typically combine with preceding or following consonants through medials or subjoined forms. Contextual variation further complicates segmentation. For instance, the word “ကို” (ko) functions as a post-particle when following a noun phrase (e.g., “ကျွန်မကို” meaning “me” as an object) but serves as an honorific title when preceding male personal names (e.g., “ကိုမင်းမင်း”). These orthographic and contextual ambiguities make Myanmar tokenization a non-trivial task.

A considerable body of research has focused on syllable-level tokenization for Myanmar text, yielding substantial progress in segmentation accuracy and computational efficiency. These advances have been supported by improved linguistic rule formulation and the creation of annotated corpora, forming the basis for downstream NLP applications such as part-of-speech tagging and machine translation.

However, despite these advances, separating syllables such as “ကျွန်”, “မ”, and “ကို” is not sufficient to capture the meaning in Myanmar context. Word-level or phrase-level text segmentation is required to better understand the context of the Myanmar language.

Previous word-level tokenization approaches have relied on dictionary-based methods and forward and backward matching algorithms, which often overlook the grammatical and contextual structure inherent in Myanmar syntax. Grammatical features such as conjunctions and post-particles play crucial roles in segmenting Myanmar sentences into meaningful words or phrases. For instance, the word or phrase preceding the post-particle “ကို” can often be identified as a noun or object, while conjunctions help to distinguish between clauses or connect multiple nouns.

Nevertheless, grammar in Myanmar is both **important and flexible**. While grammatical markers and particles provide essential structural cues, they are not

always used consistently, particularly in informal or news writing. For example, the sentence “I am not going to school today” may appear in a more common, conversational form as “ကျွန်မ ဒီနေ့ ကျောင်းမသွားဘူး။” or in a more grammatically correct form with explicit post-particles as “ကျွန်မသည် ကျောင်းကို ဒီနေ့ မသွားပါ။”. This flexibility poses additional challenges for computational models attempting to capture grammatical dependencies.

In addition, although whitespace in Myanmar orthography is neither mandatory nor standardized, it frequently conveys implicit syntactic and semantic cues. For example, in the sentence “ကျွန်မ ဒီနေ့ ကျောင်းမသွားဘူး။” (“I am not going to school today”), whitespace separates the sentence into distinct syntactic constituents: the subject (“ကျွန်မ”), temporal adverbial phrase (“ဒီနေ့”), and predicate phrase (“ကျောင်းမသွားဘူး”). This usage suggests that whitespace, though optional, can provide valuable structural information for computational processing and should be considered as an auxiliary feature in tokenization models.

To address these challenges arising from grammatical flexibility and inconsistent orthography in Myanmar, this study proposes a context-sensitive text segmentation approach that leverages grammatical and structural cues embedded in the text. Specifically, the method incorporates post-particles, conjunctions, and whitespace usage as syntactic indicators for boundary detection. These linguistic elements, combined with a set of regular-expression-based rules, enable the system to capture orthographic and morphological variations that frequently occur in Myanmar writing. In addition, the approach employs a context-aware lexicon, consisting of frequently used words categorized by grammatical function, and a tree-structured matching algorithm designed to identify the longest valid token spans during segmentation.

The key contributions of this work are as follows:

- A novel utilization of post-particles and conjunctions as structural boundary markers for Myanmar tokenization, enhancing grammatical context recognition for downstream NLP tasks.
- A systematic analysis and incorporation of whitespace as a latent syntactic cue, capturing implicit phrase and clause boundaries in natural Myanmar text.
- The development of a hybrid segmentation framework that combines hand-crafted regular-expression rules with a context-aware lexicon to achieve robust and linguistically informed tokenization.

From an artificial intelligence perspective, grammar-guided text segmentation represents a shift toward knowledge-informed natural language processing systems. While large language models increasingly dominate text processing, the interpretability of their internal representations remains limited. By embedding explicit grammatical knowledge at the foundational stage of language processing, this work advances the development of transparent and explainable AI systems.

Through the systematic integration of post-particles, conjunctions, and whitespace cues, the proposed model not only improves segmentation accuracy but also provides human-understandable reasoning for each token boundary. Such interpretability is especially crucial for low-resource languages like Myanmar, where large annotated quality corpora for training data-driven models are scarce. Moreover, the framework

establishes a linguistically grounded foundation that can later be extended with statistical or neural components to form a hybrid system—combining the transparency of rule-based methods with the adaptability of machine learning. In doing so, this research contributes to the creation of trustworthy, language-aware AI capable of processing Myanmar text with both accuracy and interpretability.

The remainder of this paper is organized as follows. Section II reviews related work, Section III describes the system architecture and methodological design, and Section IV presents experimental evaluations comparing the proposed model with baseline tokenizers.

2 Related Work

Tokenization for non-segmented and morphologically complex languages has been an active area of research in natural language processing (NLP), particularly for languages such as Thai, Lao, and Myanmar. The absence of whitespace or explicit word delimiters introduces fundamental challenges for both statistical and neural segmentation models, which often assume that token boundaries can be inferred from character frequency or co-occurrence patterns. Myanmar presents additional difficulties due to its complex orthography, including stacked consonants, variable syllable boundaries, and grammatical constructions that integrate particles, affixes, and postpositional markers.

Byte Pair Encoding (BPE), originally proposed as a data compression technique, was later adapted for NLP to mitigate out-of-vocabulary (OOV) issues by iteratively merging frequent symbol pairs [1]. BPE has demonstrated strong performance in neural machine translation (NMT), particularly for morphologically rich and low-resource languages. However, its greedy merge strategy often results in linguistically unintuitive segmentations when applied to non-segmented scripts.

Unigram Language Model tokenization, introduced by Kudo [2] and implemented in SentencePiece [3], formulates tokenization as a probabilistic model selection problem. By allowing multiple candidate segmentations during training, uni-gram tokenization has been shown to be more robust to noise and data sparsity than BPE, making it particularly suitable for low-resource settings.

To further reduce dependence on linguistic resources, character-level tokenization [4], sub-word tokenization [5] have also been explored. Character-based models eliminate explicit tokenization errors and are inherently script-agnostic; however, they typically suffer from long input sequences, increased computational cost, and difficulties in modeling long-range dependencies.

Recent work shows that tokenization choices critically affect NLP performance, especially for low-resource languages. Uzan et al. [6] demonstrate via CharBench that token granularity significantly influences downstream results, while Rust et al. [7] show that shared multilingual tokenizers systematically over-segment low-resource languages, leading to degraded performance. These findings establish tokenization as a core modeling decision rather than a mere preprocessing step.

Parallel to tokenization evaluation efforts, token-free approaches have been proposed to eliminate explicit segmentation. ByT5 [8] operates directly on UTF-8 bytes,

offering universal script coverage and robustness for low-resource and non-segmented languages. However, byte-level modeling increases sequence length and computational cost, and does not consistently outperform well-designed subword tokenizers when adequate language-specific data is available.

Multilingual pretrained language models have become a standard approach for supporting low-resource languages through cross-lingual transfer. Models such as mBERT [9] and XLM-R [10] rely on shared subword vocabularies trained across dozens of languages, enabling parameter sharing and zero-shot transfer. However, prior work has shown that this design disproportionately benefits high-resource languages, while low-resource languages suffer from vocabulary under-representation and excessive token fragmentation [7]. This issue is particularly severe for Myanmar language, where widely used subword tokenizers—including BPE-based libraries and OpenAI/GPT-style byte-level tokenization—frequently fail to produce intuitive segmentations. In practice, these tokenizers often split text at sub-syllabic or even diacritic levels, fragmenting a single syllable such as “မြန်” into individual components (e.g., “မြ”, “န”, “်”), which severely degrades downstream task performance.

In response to these limitations, recent work by Myanmar researchers has focused on language-specific word segmentation approaches. In 2025, YK. Thu [11] proposed a hybrid directed acyclic graph (DAG)-based method for Myanmar word segmentation, explicitly modeling linguistic constraints. While this approach improves word boundary detection, it does not produce tokenization at the granularity required for downstream tasks such as named entity recognition (NER). Our work is inspired by this line of research and aims to address the gap between linguistically motivated word segmentation and task-oriented tokenization for Myanmar NLP.

The framework presented in this research advances this direction by integrating linguistic knowledge directly into the text segmentation process. It builds upon prior dictionary-based efforts but extends them through a grammar-informed lexicon, a rule system encoding syntactic relationships, and a tree model that determines token boundaries through hierarchical structural analysis. This hybridization of symbolic grammar and computational modeling aligns with recent trends in explainable and language-aware AI, offering a new paradigm for Myanmar NLP that emphasizes structural understanding over statistical approximation.

3 Methodology

The proposed system introduces a grammar-guided tokenization framework that performs context-sensitive segmentation of Myanmar text using a linguistically informed, rule-based approach. Unlike purely statistical or neural models, this method explicitly integrates grammatical knowledge into the segmentation pipeline, enabling both high interpretability and robustness in low-resource conditions. The architecture combines grammar-based lexicon tagging with structured merging rules to capture syntactic relationships such as post-particle attachment, conjunction boundaries, and predicate grouping. The overall work-flow is summarized in Algorithm 1.

Algorithm 1 Grammar-Guided Tokenization Pipeline.

```
SEGMENT( $T$ )  
   $T' \leftarrow$  Protect fixed expressions  
   $S \leftarrow$  Segment  $T'$  into syllables  
   $C \leftarrow \emptyset$   
  
  for each  $s_i \in S$  do  
    if  $s_i$  is punctuation  $\rightarrow$  append Chunk( $s_i$ , PUNCT)  
    else if  $s_i$  matches pattern  $\rightarrow$  assign tag  
    else if  $s_i$  matches a lexicon  $\rightarrow$  grammatical tag  
    else label as RAW  
  
   $C \leftarrow$  clean the tags;  
   $C \leftarrow$  merge the syllables;  
   $P \leftarrow$  collapse the raw tags( $C$ )  
  return  $P$ 
```

3.1 Preprocessing and Syllable Extraction

Input text is first passed through a preprocessing module that normalizes Unicode encoding, removes extraneous whitespace, and protects fixed multi-character expressions to prevent incorrect segmentation during later stages. In this work, seven types of expressions are explicitly protected and these protected phrases are tagged accordingly:

1. **Numeric expressions:** This category includes English and Myanmar digits, numbers written in words (e.g., “တစ်ထောင်,” “သုံးဆယ်”), and phone numbers.
2. **Abbreviated organization names:** Both English and Myanmar abbreviations are identified, including those separated by special characters such as periods, hyphens, or written without separators (e.g., “U.N.,” “A.D.,” “အ.ထ.က.,” “အထက”).
3. **Dates:** Expressions denoting calendar dates in numeric or textual formats.
4. **Time expressions:** Phrases indicating specific times (e.g., “10:30 AM” or “၉:၅၀ နာရီ”).
5. **Email addresses:** Strings containing the “@” symbol followed by valid domain patterns.
6. **Named entities:** Usernames or tagged names prefixed by “@” in English or their Myanmar equivalents, as well as proper names written in *Parli* script (e.g., “သမ္မတ”).
7. **URLs:** Web addresses beginning with common protocols such as `http`, `https`, or `www`.

3.2 Lexical Resource Construction

A collection of grammatically motivated lexicons is defined and compiled into **retrieval trees or tries**, for efficient lookup during tokenization. A trie is a hierarchical tree-like data structure where each node represents a character or syllable, and a complete path from the root to a terminal node represents a valid lexical entry.

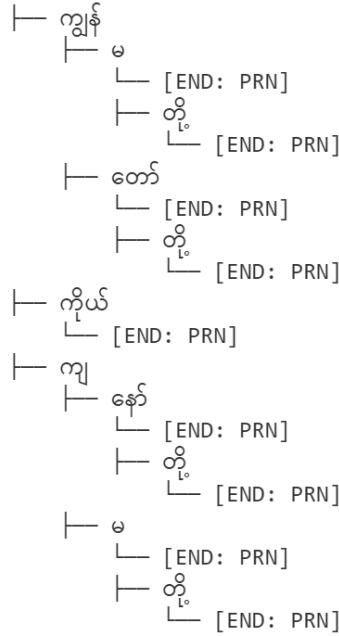


Fig. 1 An Example of Retrieval Tree Structure of the Pronounce

As illustrated in Figure 1, these trie-based lexicons provide compact storage and fast prefix-based matching for the tokenization process. This structure enables the model to efficiently search for and identify the longest matching token during segmentation, which is particularly valuable in Myanmar, where words often share common prefixes and words are written without spaces.

Each lexicon (trie) represents a distinct linguistic function in the Myanmar language, allowing the model to detect phrase boundaries and grammatical relations with high precision. The key functional lexicons are:

- **Post-particles:** markers such as “ကို,” “မှ,” “မှာ,” and “အား” that denote grammatical relations such as object, source, or location.
- **Conjunctions:** words like “နှင့်,” “သော်,” and “အတူ” that connect clauses or phrases.
- **Sentence-final particles:** discourse or mood markers such as “သည်,” “ပါ,” “တယ်,” and “လေ,” which often indicate politeness, assertion, or mood.

These three lexicons are crucial for identifying syntactic boundaries, as conjunctions, post-particles and sentence-final particles often signal the end of a phrase or clause. However, they also introduce ambiguity. In Myanmar, certain proper nouns and abbreviations contain letter sequences identical to post-particles or conjunctions. For example, “အထက” (abbreviated for “အထက်တန်းကျောင်း”) includes the syllable “က,” which is also a common post-particle, and the syllable “မ” in “ကျွန်မ” is homophonous with the particle “မ” used for negation. If un-handled, such overlaps lead to false segmentation and degradation of token boundary accuracy.

To mitigate this issue, additional noun-centered lexicons are incorporated. These lexicons serve as protective lists, allowing the model to recognize known nominal expressions as single tokens and avoid erroneous splitting by grammatical markers. They include:

- **Region names:** Major states, regions, and well-known cities in Myanmar.
- **Months and Days:** Traditional Myanmar month names, Burmese renderings of English month names, and the seven Myanmar day names.
- **Titles and honorifics:** Common Burmese titles (e.g., “ဒေါက်တာ,” “ဥက္ကဋ္ဌ,” “ဦး,” “ဒေါ်”) that frequently precede personal names.
- **Pronouns:** Core personal pronouns such as “ကျွန်မ” “ကျွန်တော်,”
- **Religious and place names:** Names of pagodas, monasteries, and other culturally significant sites.
- **Special nouns:** Frequently used organizational acronyms and consonant-only expressions (e.g., “ခန”).

These noun-based lexicons collectively provide a semantic buffer that stabilizes segmentation decisions and prevents incorrect boundary creation in cases where surface forms overlap with grammatical particles.

The model’s current focus on noun-based lexicons ensures stable, interpretable segmentation for the majority of named entities and syntactic boundaries, which are essential for downstream tasks such as named-entity recognition and part-of-speech tagging. Nevertheless, the lexicon is being incrementally expanded to include verbs as more annotated resources become available. This progressive integration will ultimately allow the system to handle predicate-level segmentation, improving its ability to capture complete syntactic and semantic units in Myanmar text.

In the current implementation, however, the text segmentation engine primarily relies on the existing noun-based and grammatical lexicons, supported by a rule-based segmentation mechanism. To operationalize these linguistic resources efficiently, the model employs a retrieval-tree (trie) structure for fast lookup and context-sensitive matching during the segmentation process.

3.3 Trie Scanning and Longest-Match Search

All lexicons are compiled into **retrieval tree structures** within the processing pipeline for rapid and prioritized pattern matching. The pipeline performs a longest-match search across the defined lexicons, following the grammatical hierarchy encoded in the PIPELINE configuration. High-priority lexical categories such as regions, months, and titles, are evaluated before functional markers like post-particles and

sentence-final particles. This ordering ensures that known nominal expressions are recognized as atomic units before grammatical particles are applied for boundary detection.

Once the input text has been syllabified, the segmentation module performs a longest-match search across all lexicons in the processing pipeline. The scanning process begins by selecting the token at position i as the starting node in the trie. It then descends through successive trie branches, one token at a time, matching each against the children of the current node. At each level, if a branch terminates in a node marked as end-of-entry, the algorithm records that span as a potential match.

However, it does not immediately return; instead, it continues traversing deeper into the trie to determine whether a longer valid match exists. When no further branch corresponds to the next token in the sequence, the traversal stops, and the longest recorded match is selected as the output for that position. If multiple lexicons yield overlapping matches starting at the same token, the algorithm applies a priority rule:

- Matches from lexicons appearing earlier in the pipeline (i.e., higher grammatical priority) are preferred.
- Among these, the longest span (i.e., the match covering the most tokens) is selected.

The result is encoded as a structured unit called a Chunk, which stores the token span (i, j) , the concatenated text of the matched sequence, and the corresponding grammatical tag (e.g., POSTP, NOUN, CONJ). If no valid match is found at position i , the token is labeled as RAW.

3.4 Post-particle Cleaning and Re-labeling

Following initial tagging, a targeted cleaning stage is applied to refine the boundaries generated by the grammatical lexicons. In Myanmar, post-particles such as “ကို,” “မှ,” “မှာ,” and “အား” are reliable indicators of syntactic relations; however, certain post-particles (e.g., “က,” “ကို,” “မ,” “လ”) also frequently appear as part of personal names or titles, leading to incorrect segmentation if left uncorrected. To address this issue, the system applies a post-particle re-labeling mechanism that adjusts token tags based on local punctuation and positional context.

When a token identified as a post-particle violates expected syntactic positioning such as appearing at the beginning of a sentence or directly adjacent to punctuation, it is reclassified as a neutral element rather than a grammatical marker. This adjustment prevents the tokenizer from misinterpreting these surface forms as legitimate syntactic boundaries, thereby improving segmentation accuracy in both formal and informal Myanmar text.

3.5 Structural Merging

After the cleaning phase, the tokenizer performs a series of merging operations to reconstruct meaningful multi-syllable expressions. These merges operate sequentially to consolidate tokens that belong to the same grammatical or semantic phrase, ensuring that segmentation respects Myanmar’s morphosyntactic structure.

Merging between boundaries.

The first merging stage identifies tokens that occur between major grammatical boundaries, typically delimited by conjunctions, post-particles, or sentence-final particles and merges them into coherent phrasal units. This operation reflects the fact that content words and modifiers often occur in contiguous sequences bounded by functional markers. By merging tokens within these spans, the system forms linguistically interpretable constituents that align with natural syntactic groupings.

Merging numeric expressions.

To preserve semantic transparency, the model merges only consecutive numeric material, including Burmese/Arabic digits and Myanmar number words, into a single numeric token, while leaving classifiers and measurement units as separate tokens. This design avoids over-aggregation, maintains the syntactic role of classifiers, and supports downstream tasks (e.g., agreement, counting, or unit-aware reasoning) that rely on recognizing NUM-CL structure. Crucially, protected formats (dates, times, phone numbers, and decimals) are handled upstream and are not altered by this merge.

Predicate-level merging.

The final merging phase targets predicate structures, combining verbs with their auxiliary particles or negation markers (e.g., “သွားမယ်,” “မသွားဘူး”). This step captures the syntactic and semantic completeness of verb phrases, which often express tense, modality, or polarity through auxiliary morphemes rather than inflection. By merging these elements into cohesive predicate units, the tokenizer preserves the functional boundaries essential for clause-level analysis and downstream tasks such as dependency parsing or machine translation.

4 Results

The performance of the proposed grammar-guided tokenization framework was evaluated using a manually annotated corpus designed to capture both the formal and informal characteristics of contemporary Myanmar text. The corpus contains 1,034 Myanmar sentences drawn from both formal and informal domains. The evaluation corpus comprises three primary datasets:

- **News Articles (Mixed Formality):** 293 sentences collected from diverse Myanmar news outlets. While some news sources maintain formal written conventions, others adopt more conversational or colloquial expressions, frequently omitting post-particles, using unstandardized spacing, or mixing spoken forms with written syntax. This dataset provides a realistic benchmark for assessing segmentation robustness under varying degrees of grammatical adherence.
- **Government Texts (Formal Domain):** 209 sentences and statements drawn from government-issued textbooks and official press communications. These texts adhere closely to formal grammatical conventions, exhibiting explicit use of post-particles, standardized phrasing, and well-structured syntax. This subset serves as a reference domain for evaluating segmentation performance under grammatically precise and orthographically consistent conditions.

- **Novels (Formal but Stylistically Variable)**: 532 sentences, randomly extracted from Myanmar novels written by three different authors. Although these texts generally follow formal grammatical conventions, they involve dialogue and exhibit stylistic variation. This dataset allows evaluation of segmentation performance in formal contexts that nevertheless display natural stylistic diversity.

To ensure both reliability and linguistic validity, all text samples were manually segmented following a standardized annotation protocol. Four annotators, including one Myanmar language teacher, were instructed to determine word and phrase boundaries based on grammatical and semantic coherence rather than orthographic spacing alone. Boundaries were marked at the edges of meaningful lexical or functional units, including post-particles, conjunctions, sentence-final particles, and complete nominal or verbal phrases. Elements protected during preprocessing, such as URLs, numerals, and abbreviations, were annotated as single, indivisible tokens.

The evaluation focused exclusively on meaningful **tokenization** rather than tagging. Its primary goal was to assess whether the proposed methodology can accurately segment sentences into grammatically and semantically meaningful phrases, even when the input text is informal, ungrammatical, or limited in lexical coverage. The current lexicon contains less than 1000 entries, including particles and conjunctions. Evaluation was carried out across three principal dimensions:

- **Intrinsic** segmentation quality relative to human annotations
- **Extrinsic** (downstream) evaluation via a named entity recognition (NER) task using a baseline model
- **Qualitative error analysis**, aimed at identifying key linguistic and orthographic challenges that contribute to segmentation errors.

The following subsections present detailed results and analyses for each dimension.

4.1 Intrinsic Segmentation: Boundary-level Evaluation

This subsection reports the core evaluation results for boundary detection. Performance was assessed using standard intrinsic segmentation metrics, which measure alignment between system-predicted and gold-standard token boundaries:

- **Boundary Precision (P)** — the proportion of predicted boundaries that correctly match gold boundaries.

$$P = \frac{|B_{\text{pred}} \cap B_{\text{gold}}|}{|B_{\text{pred}}|} \quad (1)$$

- **Boundary Recall (R)** — the proportion of gold boundaries correctly identified by the system.

$$R = \frac{|B_{\text{pred}} \cap B_{\text{gold}}|}{|B_{\text{gold}}|} \quad (2)$$

- **Boundary F1 Score (F1)** — the harmonic mean of Precision and Recall.

$$F1 = \frac{2PR}{P + R} \quad (3)$$

Table 1 below summarizes the segmentation performance of the model across three different test datasets: government statements, novels, and news. Each dataset is evaluated in terms of precision, recall, and F1 score.

Dataset	Test Size	Precision	Recall	F1 Score
gov-statements	209	0.8720	0.9954	0.9296
novels	532	0.8767	0.9840	0.9272
news	293	0.9215	0.9219	0.9217

Table 1 Model performance across Data-sets (Boundary Segmentation)

As shown in Table 1, the F1 scores across all three datasets are very close (approximately 0.92–0.93), indicating that the proposed method is robust across different text domains despite variations in writing style, sentence structure, and vocabulary.

For the gov-statements and novels datasets, recall is particularly high (0.9954 and 0.9840), while precision is comparatively lower (approximately 0.87). This pattern suggests that the model is highly sensitive to segmentation boundaries and rarely misses true boundaries, albeit at the cost of introducing additional false positives.

In contrast, for the news dataset, precision and recall are closely aligned (approximately 0.92), resulting in a slightly lower but more balanced F1 score. This indicates that news text exhibits more regular or predictable boundary patterns, enabling the model to avoid over-segmentation.

Despite variations in test set size (ranging from 209 to 532 instances), overall performance remains stable across datasets, suggesting that the observed differences are primarily driven by text characteristics rather than dataset scale.

4.2 Intrinsic Segmentation: Token-level Evaluation

In addition to boundary-based evaluation, token-level accuracy was measured to assess the correctness of the resulting tokens as complete units. A token is considered correct if it exactly matches a reference token in both span and content. Token-level evaluation provides a more holistic measure of segmentation quality by comparing the predicted tokens directly against the gold standard, rather than focusing solely on boundary alignment. Accordingly, we report the following metrics:

- **Precision** — the proportion of predicted tokens that exactly match reference tokens
- **Recall** — the proportion of reference tokens that are correctly recovered by the model
- **F1- score** — the harmonic mean of precision and recall

This intrinsic evaluation complements boundary-based metrics and serves as a prerequisite for downstream extrinsic evaluation. Table 2 presents the token-level intrinsic segmentation performance of the model across three data-sets: government statements, novels, and news. Recall scores are consistently high, ranging from 0.8156 to

0.8725, with the highest recall observed on the government statements dataset, indicating effective recovery of gold-standard tokens despite occasional over-segmentation. Precision ranges from 0.7692 to 0.8153.

Dataset	Test Size	Precision	Recall	F1 Score
gov-statements	209	0.7692	0.8725	0.8176
novels	532	0.7792	0.8696	0.8219
news	293	0.8153	0.8156	0.8155

Table 2 Token-level segmentation performance across datasets

4.3 Extrinsic Evaluation: Named Entity Recognition

Table 3 reports the extrinsic evaluation results of the proposed segmentation approach using a named entity recognition (NER) task. The tokenizer output is provided as input to a fixed NER model, enabling an assessment of how segmentation quality influences downstream sequence labeling performance. NER is particularly sensitive to token boundaries, as segmentation errors may cause entity fragmentation or incorrect merging, directly affecting recognition accuracy.

Performance is evaluated using standard entity-level precision, recall, and F1-score across five entity categories: location (LOC), number (NUM), person (PER), DATE, and TIME. Results obtained with the proposed segmentation method are compared against those produced using a baseline tokenization strategy while keeping the NER model architecture unchanged. This setup ensures that performance differences primarily reflect the impact of tokenization rather than model design.

Method	Metric	LOC	NUM	PER	DATE	TIME
Proposed method	Precision	0.6946	0.7237	0.8209	0.8148	0.8000
	Recall	0.6629	0.6044	0.6250	0.7586	0.7500
	F1-score	0.6784	0.6587	0.7097	0.7857	0.7742
Baseline	Precision	0.9082	0.5492	0.6790	0.8873	0.5263
	Recall	0.9207	0.4379	0.5140	0.8630	0.4444
	F1-score	0.9144	0.4873	0.5851	0.8750	0.4819

Table 3 NER Performance on Tokenized Outputs of the Baseline and Proposed Methods

It is important to note that this comparison is not strictly apple-to-apple. The NER model was trained using ground-truth annotations aligned with the baseline tokenization ([11]) and was not retrained for the proposed segmentation scheme. Consequently,

the results should be interpreted as an indication of how well the proposed tokenizer generalizes to an existing downstream NER system without additional retraining.

Without retraining the downstream model, the proposed method demonstrates strong performance for several entity types, achieving F1-scores of 0.71, 0.79, and 0.77 for PER, DATE, and TIME, respectively (Refer to Figure 3). Slightly lower performance is observed for LOC and NUM entities, which can be attributed to differences in how contextual modifier words are handled by the grammar-based tokenization compared to the ground-truth annotations used in NER training. Specifically, the proposed method merges supportive words surrounding an entity into a single token based on grammatical rules. For example, in the phrase “ဘုတလင် | မြို့ | မြောက်ဘက်”, the town (မြို့) and the directional modifier (မြောက်ဘက်, meaning north of) are merged into a longer token by the proposed tokenizer. In contrast, in the NER ground truth and baseline tokenization, “ဘုတလင် | မြို့” is labeled as a location entity, while directional words are treated as outside the entity span.

A similar behavior is observed for numeric expressions. For instance, the phrase “သုံးယောက်ခန့်” is treated as a single token by the grammar-based tokenizer, as the suffix “ခန့်” indicates approximation. However, in the NER ground truth, only “သုံးယောက်” is labeled as a numeric entity, with the approximation marker considered outside the entity span. Such tokenization mismatches lead to under-recognition when evaluated against a model trained with baseline token boundaries.

Although the BERT-based NER model is capable of handling over-segmented text through sequential prediction, similar to the baseline setting, it remains limited in its ability to recover entity boundaries within a single merged token. Consequently, grammar-driven over-merging may adversely affect recognition for certain entity types, particularly those that are frequently accompanied by contextual modifiers.

Nevertheless, the proposed method demonstrates competitive and stable performance across multiple entity categories. In particular, it maintains strong results for PER, DATE, and TIME entities, indicating that the segmentation strategy effectively preserves meaningful token boundaries for temporally and semantically structured expressions. In addition, the proposed approach exhibits a more balanced trade-off between precision and recall for NUM entities compared to the baseline, achieving an F1-score of 0.6587 versus 0.4873. Taken together, these results suggest that the proposed tokenizer can be effectively applied to downstream NER tasks even without retraining the underlying model, highlighting its practical potential for real-world NLP applications.

4.4 Error Analysis

To understand the residual challenges of the proposed tokenization approach, we conducted an error analysis on 500 mis-segmented sentences sampled from three different datasets.

4.4.1 Role ambiguity induced segmentation errors.

This error category is caused by Burmese particles (e.g., ‘သည်’ and ‘၏’) that exhibit dual roles as both syntactic markers and sentence-final particles. In such cases, a

single segmentation strategy cannot simultaneously preserve clean entity boundaries and ensure correct attachment of sentence-final particles to verbs. While position-aware heuristics are applied during preprocessing, verb particle sentence endings may appear not only at the end but also at the beginning or in the middle of a sentence, especially in informal and novel-style texts.

As a result, these ambiguities cannot be fully resolved without higher-level syntactic information (e.g., whether the preceding token is a noun or a verb), leading to the residual errors illustrated below.

Tokenizer Output	Gold Segmentation	Explanation
လေယာဉ်ပျံကြီးသည်	လေယာဉ်ပျံကြီး သည်	Ambiguous particle attachment to the preceding nominal token
မှတ်ချက်ချ၊ သည်	မှတ်ချက်ချသည်	Incorrect splitting of a verb-particle construction

Table 4 Representative segmentation errors caused by particle role ambiguity.

4.4.2 Lexicalized numeral-quantifier expressions.

In general, segmenting Burmese numerals and classifiers into separate tokens is desirable for downstream tasks such as named entity recognition, where numerals and classifiers are treated as distinct semantic units. Accordingly, the proposed method intentionally splits numeral-classifier constructions.

However, a subset of numeral-quantifier expressions in Burmese are lexicalized and function as idiomatic units whose meanings are not strictly compositional. For example, expressions such as ‘တစ်စုံတစ်ခု’ (“something”) and ‘တစ်ခါမျှ’ (“ever” or “once”) convey abstract or emphatic meanings that differ from the literal interpretation of their constituent morphemes. Representative examples are shown in Table 5.

Tokenizer Output	Gold Segmentation	Explanation
တစ် စုံ တစ် ခု	တစ်စုံတစ်ခု	Lexicalized indefinite expression
တစ် ခါ မျှ	တစ်ခါမျှ	Lexicalized emphatic temporal expression

Table 5 Examples of lexicalized numeral-quantifier expressions that are segmented compositionally by the tokenizer.

Since the proposed tokenizer does not rely on a predefined lexicon of such lexicalized expressions, an intentional design choice to maintain generalization, (though we plan to incorporate the commonly used expressions in the future version) these cases are segmented, leading to residual errors when compared against gold segmentations that treat them as single units.

4.4.3 Particles in compound verbs and nouns.

This error category arises from Burmese particles (e.g., ‘အား’) that form part of compound verbs or nouns. Since the proposed method does not explicitly rely on a large predefined lexicon of Burmese compound words (an intentional design choice), given the productivity of compounding and the potential for meaning shifts depending on particle attachment such compounds are often segmented into smaller units.

As a result, these ambiguities cannot be fully resolved without constructing a lexicon that enumerates particle-based compounds with potentially ambiguous meanings, leading to the residual errors illustrated below.

Tokenizer Output	Gold Segmentation	Explanation
အား ယူ	အားယူ	Incorrect splitting of a compound verb
လော က	လောက	Erroneous decomposition of a lexicalized noun into subcomponents

Table 6 Representative segmentation errors caused by particles in compound verbs and nouns.

4.4.4 Adjectival constructions

This error category arises from the adjectival use of the Burmese markers သော and သည့်, which function as relativizers that transform verbs or clauses into adjectival modifiers. Although these markers are sometimes loosely described as conjunctions, in such constructions they serve a derivational role by forming adjective-like phrases rather than linking independent clauses.

In Burmese, expressions such as ဖြစ်နေသော, လိုအပ်သည့် are commonly treated as single adjectival units modifying a following noun. However, the proposed method treats သော and သည့် as general-purpose functional particles and segments them separately to preserve compositionality and generalization across domains. Representative examples are shown in Table 7.

Tokenizer Output	Gold Segmentation	Explanation
ဖြစ်နေ သော	ဖြစ်နေသော	Incorrect splitting of an adjectival modifier formed by a bound relative marker
လိုအပ် သည့်	လိုအပ်သည့်	Erroneous separation within an adjectival construction using an attributive marker

Table 7 Representative segmentation errors in adjectival constructions.

As a result, these adjectival constructions are often segmented into smaller units, leading to residual errors when compared against gold segmentations that treat them as unified modifiers. While a possible design change would be to model သော and သည့် as special adjectival markers, doing so would require explicit handling of relativized clauses and reduce the generality of the tokenizer.

5 Discussion

The experimental results provide complementary insights into the strengths and limitations of the proposed grammar-based tokenization approach. Intrinsic evaluation demonstrates that the method produces stable and linguistically motivated segmentations across diverse Burmese text sources, indicating its effectiveness in capturing common grammatical patterns. These results suggest that the proposed approach is well aligned with the structural properties of the language and offers a consistent segmentation strategy across datasets.

The extrinsic evaluation using a named entity recognition (NER) task further shows that the proposed tokenization generalizes reasonably well to downstream applications. Even though the NER model was trained using baseline tokenization and was not retrained for the proposed scheme, competitive performance is achieved across several entity categories, particularly for PER, DATE, and TIME entities. This robustness under tokenization mismatch highlights the practical applicability of the proposed tokenizer in real-world scenarios, where retraining downstream models may not always be feasible.

To better understand the remaining errors, a qualitative error analysis was conducted on 500 mis-segmented sentences sampled from three datasets. The analysis reveals that a significant portion of residual errors stems from role ambiguity induced by Burmese particles, such as “သည်” and “၏”, which can function both as syntactic markers and as sentence-final particles. In these cases, a single segmentation strategy cannot simultaneously preserve clean entity boundaries and ensure correct attachment of sentence-final particles. Although position-aware heuristics help mitigate some of these issues, such ambiguities cannot be fully resolved without access to higher-level syntactic information or large lexicon, particularly in informal or narrative-style text where sentence-final particles may appear in non-canonical positions.

Overall, these findings suggest that while grammar-based tokenization provides a strong and interpretable foundation for Burmese text processing, further improvements may require incorporating syntactic or semantic cues beyond surface-level grammatical rules. Nonetheless, the proposed approach strikes a favorable balance between linguistic interpretability and downstream performance, making it a practical solution for low-resource language settings and a promising basis for future hybrid or syntax-aware tokenization methods.

6 Future Work

Future work will focus on extending the proposed grammar-guided framework into a hybrid tokenization model that integrates linguistic rules with data-driven learning techniques. While the current approach offers transparency and interpretable segmentation, it relies primarily on handcrafted rules and lexical patterns. Incorporating data-driven learning components, such as sequence labeling models, subword-based tokenization, or lightweight transformer-based architectures, would enable the system to automatically infer token boundaries from contextual usage while retaining the grammatical interpretability of the rule-based layer.

In this hybrid design, linguistic constraints derived from post-particles, conjunctions, and syntactic patterns could serve as soft priors or attention guides during model training, allowing the neural component to generalize beyond explicitly defined rules. This integration has the potential to address residual errors identified in the error analysis, particularly those arising from particle role ambiguity, while improving robustness to dialectal, informal, or noisy text. Ultimately, this progression toward a grammar-informed neural framework aims to balance adaptability and interpretability, supporting the development of effective NLP systems for Myanmar and other low-resource languages.

7 Conclusion

This work presents a grammar-guided tokenization approach for Myanmar that leverages linguistic knowledge to produce interpretable and consistent segmentation. Through intrinsic evaluation, we demonstrate that the proposed method generates stable token boundaries aligned with grammatical structure across diverse datasets. Extrinsic evaluation using a named entity recognition (NER) task further shows that the proposed tokenizer generalizes effectively to downstream applications, achieving competitive performance across multiple entity categories in downstream NER evaluation even when applied to a model trained with a different tokenization scheme.

A detailed error analysis reveals that most residual errors stem from inherent linguistic ambiguities in Myanmar, particularly the dual roles of post-particles and sentence-final markers, which are difficult to resolve using surface-level rules alone. Despite these challenges, the proposed approach maintains robust performance and offers clear interpretability, making it a practical solution for low-resource language settings where annotated data and retraining opportunities are limited.

Overall, this study highlights the value of incorporating grammatical knowledge into tokenization for Myanmar and demonstrates that rule-based approaches can remain effective and competitive in modern NLP pipelines. The findings lay a solid foundation for future hybrid tokenization methods that combine linguistic constraints with data-driven learning to further improve adaptability and performance.

Acknowledgements. The authors acknowledge the Myanmar language teachers for their valuable assistance in understanding the Myanmar (Burmese) language.

Declarations

Conflict of interest

The authors declare that they have no conflicts of interest to disclose.

Data and Code availability

The grammar-based tokenization method is released as an open-source Python library and is publicly available via PyPI at <https://pypi.org/project/mmdt-tokenizer/0.1.2/>. The source code is also accessible on GitHub at <https://github.com/Myanmar-Data-Tech/mmdt-tokenizer>.

References

- [1] Sennrich, R., Haddow, B., Birch, A.: Neural machine translation of rare words with subword units. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, pp. 1715–1725 (2016)
- [2] Kudo, T.: Subword regularization: Improving neural network translation models with multiple subword candidates. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, pp. 66–75 (2018)
- [3] Kudo, T., Richardson, J.: SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Brussels, Belgium, pp. 66–71 (2018). <https://doi.org/10.18653/v1/D18-2012>
- [4] Cherry, C., Foster, G., Bapna, A., Firat, O., Macherey, W.: Revisiting character-based neural machine translation with capacity and compression. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp. 4295–4305 (2018). <https://doi.org/10.18653/v1/D18-1461>
- [5] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI Technical Report (2019)
- [6] Uzan, O., Pinter, Y.: CharBench: Evaluating the Role of Tokenization in Character-Level Tasks (2025). <https://arxiv.org/abs/2508.02591>
- [7] Rust, P., Pfeiffer, J., Ruder, S., Gurevych, I.: How good is your tokenizer? On the monolingual performance of multilingual language models. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (2021)
- [8] Xue, L., Barua, A., Constant, N., Al-Rfou, R., Narang, S., Kale, M., Roberts, A., Raffel, C.: ByT5: Towards a token-free future with pre-trained byte-to-byte models. Transactions of the Association for Computational Linguistics **10**, 291–306 (2022) https://doi.org/10.1162/tacl_a_00461
- [9] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT (2019)
- [10] Conneau, A., Khandelwal, K., Goyal, N., *et al.*: Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (2020)
- [11] Thu, Y.K.: oppaWord: Hybrid DAG+Bi-MM+LM Myanmar Word Segmenter (2025). <https://github.com/ye-kyaw-thu/oppaWord>