International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Computational Implementation of Morphological Analysis and Generation of Verbs in Myanmar Language

***Kaung Myat Thu [1], H. Mamata Devi [2], Th. Rupachandra Singh [3]**

Submitted: 06/10/2023 **Revised**: 21/11/2023 **Accepted**: 02/12/2023

Abstract: The field of morphological analysis and generation focuses on the study of word production, the recognition of grammatical components within words, and the creation of words that adhere to morphotactic standards. According to various research reports, finite-state techniques are fast, effective, and efficient in interpreting human language morphologies into the computational system. FOMA: a more elaborate version of Xerox's finite state toolset can be used to implement the finite state morphology. Using FOMA toolset and other programming languages, we have already created the MM-Morph tool: a computational linguistic tool for morphological analyzer and generator for Myanmar nouns. In this paper, we describe the linguistic phenomena of the morphology of verbs and the techniques used in the system's development process to integrate it into the existing MM-Morph tool. MM-Morph has been developed as a part of the research "Morphological Analysis and Generation for Myanmar Language using Finite State Techniques." We share the experimental evaluations conducted to assess this system's performance. Evaluation results show that the MAG system of Myanmar verbs can identify more than (78%) of the verbs in the language.

Keywords: *Morphology, Natural Language Processing, FST, MM-Morph, FSA, Morphological analysis and generation (MAG), finite-state morphological analysis and generation (FSMAG), Computational linguistics tool, Indian Language, Myanmar Language, Xerox's xfst, FOMA*

1. Introduction

Myanmar, also known as Burmese, is the official language of Myanmar (formerly known as Burma) and is spoken by the majority of the country's population. It belongs to the Sino-Tibetan language family and is part of the Tibeto-Burman branch. Myanmar's unique script, the Burmese script, is used for writing the language. The history of the Myanmar language is rich and complex, spanning centuries of cultural, political, and linguistic evolution. This paper presents a dedicated effort toward creating a computational linguistic tool designed to analyse the morphology of the Myanmar language, with a specific emphasis on its verb patterns.

Morphological Analysis and Generation (MAG) is a critical step in any natural language processing task, including speech recognition, lemmatization, POS tagging, spell checking, grammar checking, machine translation, text summarization, word sense disambiguation, information extraction, and search engines. Due to its agglutinative structure and intricate morphological occurrences, developing computational linguistic tools for the Myanmar language is complicated. We have already created the MM-Morphs, a computational linguistic tool for morphological analysis and generation of Myanmar nouns. In this paper, we describe the linguistic phenomena of the morphology of verbs and the techniques used in the development process to integrate it into the existing MM-Morph tool. MM-Morph combines the two critical computational linguistic tools to identify the root words, affixes, and grammatical components inside words to generate new words, i.e., their lexical forms and surface forms.

The paper is structured into the following successive sections. We will briefly discuss the related efforts made by researchers for the different languages in section (II). We describe the theoretical concepts of our approach in section (III). We explore some linguistic aspects of the morphology of Myanmar verbs in section (IV). We present the implementation of the system and its integration into MM-Morph in section (V). In section (VI), we evaluate the system's performance. The study is concluded with a few summaries of the related works in section (VII).

2. Existing Approaches

Many approaches are widely used in morphological analysis. A brief description of commonly used methods is described as follows:

Corpus-Based Approach: This statistical approach relies on a substantial corpus as its training data. Employing a suitable machine learning algorithm, the system is trained to extract vital information and features from the corpus, subsequently utilizing this knowledge to assess new data. The primary challenge lies in constructing an annotated corpus. [2], [3], [4], [5], [6], [7], [9], [10]

*[1] Research Scholar Department of Computer Science, Manipur University
[2] Professor Department of Computer Science Manipur University
[3]Assistant Professor MCA (Manipur University)

Paradigm-Based Approach: Within a given language, words falling under distinct categories, such as nouns, verbs, adjectives, adverbs, and postpositions, can be categorized into specific paradigms based on their morphophonemic behavior. A morphological analyzer, utilizing a paradigm-based morphological compiler program, is developed based on the paradigms or sets of every word. [7], [8]

Finite State Automata (FSA) Based Approach: This approach employs regular expressions to accept or reject a string within a designated language. FSA analyzes the system's behavior, comprising states, transitions, and actions. Upon initiation, FSA operates in an initial stage, ceasing work if it reaches any of the final states. [13], [14], [15],[21], [22], [23], [24], [25], [26].

Two-Level Approach: Kimmo Koskenniemi's 1983 development, known as Two-level morphology, revolutionized word-form recognition and generation. It involves a computational model with two levels — lexical and surface form. Representing a word as a direct, letter-for-letter correspondence between these forms, the approach is characterized by three key concepts:

- Rules operate as symbol-to-symbol constraints applied concurrently, not sequentially like rewrite rules.
- Constraints may reference lexical context, surface context, or both simultaneously.
- The process of lexical lookup and morphological analysis takes place concurrently [2].

Directed Acyclic Word Graph (DAWG) Approach: DAWG is capable of storing interconnected word forms, lemmas, grammatical information, and morphological structure of words. The structure is like a lexical transducer with two layers called surface and lexical forms as well as morphological information. It can store finite strings of information in a compressed way. Thus, MAG can be processed very fast even if many parts of the information couldn't be searched from the input. The presented approach is language-independent, and the system does not require any other special linguistic information as well as morphotactic rules. [28]

Stripping Approach: Especially effective in highly agglutinative languages like Dravidian languages, this method capitalizes on the frequent occurrence of words formed by appending suffixes to root words. Identifying the suffix allows extraction of the word stem by applying appropriate orthographic rules. [29], [30].

3. Finite-State Technology – The Theoretical Framework

Finite-state transducer (FST) has been the subject of extensive research and interest for the past almost fifty years. Its mathematical elegance and efficient implementation have been well documented in the vast literature available. Finite state transducers extend the concept of finite state machines by associating output symbols with transitions. In the context of FSMAG, FSTs are used to map between word forms and their corresponding grammatical representations.

In particular, an FST can be used to model the process of analyzing a word and determining its morphological components (e.g., roots, affixes, etc.), as well as generating a word from its morphological components. In this context, a six-tuple FST is a specific type of FST that includes six elements: $T = (Q, \theta, \Gamma, \delta, s, \gamma)$ where Q is a finite set of states (e.g., Q= {q0, q1, q2, q4,}), θ is a finite set of the input string, (or) word form for Myanmar words (e.g., θ= {ကြည့်, စား,ချစ် …}), Γ is a finite set of output string, (e.g., Γ = {+V, +PAST, …}), $\delta:(Q, \theta): Q \times \theta \rightarrow Q$ is the transition function, $s \in Q$ is the start state, and $\gamma: \in Q$ is the final state.

Example: Converting Myanmar Verbs to Past Tense

- Q: Two states, S0 and S1.
- θ: Input symbol, like the base form of a verb (e.g., "ချစ်").
- Γ: Output symbol, like the past tense of a verb (e.g., "ချစ်ခဲ့").
- δ: Transition rules: (S0, base_form) -> (S1, past_tense).
- s: Starting state, S0.
- γ: Final state, S1.

Process:

1. Start in state S0.
2. Input "ချစ်" (base_form).
3. According to the rule, move to state S1 and output "ချစ်ခဲ့" (past_tense).

The six-tuple FST defines the rules and steps to transform a base verb to its past tense. Collectively, these six components encapsulate the behavior and operation of the FST, allowing it to transform input symbols into corresponding output symbols through a sequence of state transitions defined by the transition function δ. This formal representation enables the FST to model various linguistic and computational processes, making it a powerful tool for analyzing and generating language-related patterns.

A mathematical model of the morphotactics of the Myanmar language as a finite set of input symbols can be presented as the following formula.

$\theta = \{\sum_{i=0}^{1} px_i\} + R + \{\sum_{i=0}^{1} px_i\}$, where θ represents the input word form, R for the root word, px for prefixes, and sx stands for suffixes. The transduction process maps surface forms to their corresponding lexical forms for morphological analysis. For instance, in the case of analysis, it takes input (စားခဲ့တယ်/had eaten) as a surface form and which is transformed into a lexical form (စား+PAST+REAL).

Table 1: Some examples of FST results according to the word formation algorithm

Lexical form	Surface	Type
ချစ်+V +PAASP +PL+PERASP	ချစ်ခဲ့ကြပြီး /had loved	Inflection Morphology
စား+V +PAASP +PL+PERASP+ ADJVAL	စားနေခဲ့ကြရင်း /while eating	Derivational Morphology / Remarks: Change Verb to Adverb in a complex morphotactics
ချစ်+NOM	ချစ်ခြင်း /loving	Derivational Morphology/word with one suffix Remarks: change verb to noun
လှပ+V+PAASP+ ADJVAL	လှပခဲ့သော/ beautiful	Derivational Morphology/ Remarks: change verb to adjective, in this case the word "လှပ" can be assumed as adjective.
ချစ်+V+ADVAL + ADJVAL	ချစ်စွာသော/ beautifully	Derivational Morphology/ Remarks: contains two suffixes, 1st suffix change verb to adverb, and then 2nd suffix change adverb to adjective again

4. Morphology of Myanmar Verbs

The morphology of Myanmar verbs exhibits a complex and intriguing structure, reflecting the language's rich linguistic heritage and unique features. Myanmar, also known as Burmese, is an agglutinative language, meaning that various morphemes are added to a root word to convey different grammatical meanings. This is particularly evident in the structure of its verbs. In Myanmar, verbs are typically composed of several morphemes that indicate tense, aspect, mood, and other grammatical nuances. Compared to noun morphology, verb morphology is more complex. Words in the Myanmar language can be simply created with affixations. The Myanmar language does not have a distinct word class except for nouns. Some root words can be transformed into verbs, adjectives, and adverbs by attaching affixes.

Table 2: Some examples of verb root words followed by suffixes.

No	Surface Form	Lexical Form	Word Formation
1	အချစ်	+NZR+ချစ်+V+N	px1→R→sx0
	ချစ်ခဲ့သည်	ချစ်+PAASP+REAL	
2	ချစ်သည်	ချစ်+SASP+REAL	px:0→R→sx1→sx2
	ချစ်နေသည်	ချစ်+PRGASP+REAL	
3			px:0→R→sx1
4			px:0→R→sx1→sx2

The word classes of the word forms change depending on the choice of affixes. Most verbs have a monosyllabic root, and the primary way that verb phrases are processed is by adding suffixes to the root to give the manner of action, process, tense, aspect, intention, politeness, mood, etc. If a verb has no suffix attached, then it is in imperative commands. For instance, the aspect marker "-ထား" (-htar) can be added to the verb "ပြောထား" (pyau-htar), forming "ပြောထားသည်" (pyau-htar the), which means "have already talked." Here, the aspect marker adds the sense of perfect ready action to the verb root. Moreover, the complexity of Myanmar verb morphology extends to the inclusion of mood markers. Consider the verb "သွား" (thwar), meaning "to go." By adding the mood marker "-မှာ" (-hma) to it, we create "သွားမှာ" (thwar hma), signifying "will go anyway." This showcases how mood is intricately expressed through affixation in Myanmar verbs. This complexity adds depth and richness to the language, allowing speakers to convey a wide range of meanings within a single verb phrase. The number of verbal suffixes is much more than the nominal suffixes. The following diagram shows the verbal affixes database structure and their lexical notation.

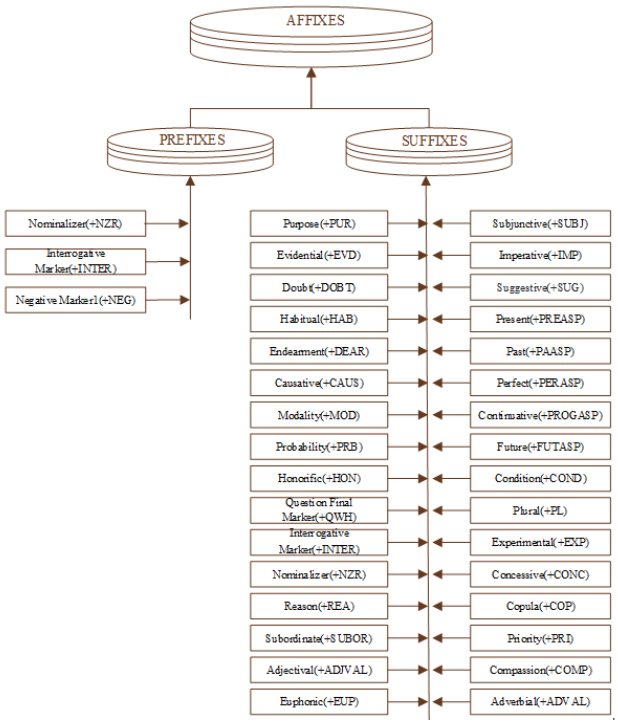

Fig. 1: Design and features of affix dictionary.

5. Development Process

5.1 Lexicon

To construct the system, a prerequisite involves acquiring, at minimum, a lexicon encompassing the compilation of stems and affixes accompanied by fundamental information and morphotactics. A lexicon is a repository for words. The lexicon of a language refers to its vocabulary, encompassing an exhaustive collection of all the words within that language. Enumerating each individual word in a language can be an uphill task. Therefore, computational lexicons are commonly employed to serve this purpose more efficiently. Collecting stems and affixes(prefixes and suffixes) from mypos corpus, English-Myanmar Dictionary database and some Myanmar Grammar books has been done semi-manually. Subsequently, distinctive terms are gleaned from the corpus, systematically arranged to facilitate manual word processing. These identified words are then categorized into different classes through manual classification. Our main lexicon contains, actually, 7800 of verbs and 145 affixes represented as the manner of action, process, tense, aspect, intention, politeness, mood, etc.

5.2 Morphotactics

Morphotactics pertains to the arrangement or ordering of morphemes within a word. This conceptual model elucidates the various classes of morphemes present within the structure of a word.. The morphemes of a word can never appear in a random sequence. A set of rules known as morphotactics precisely limits the sequence in which morphemes follow one another. In agglutinative languages like the Myanmar language, words are generated by

concatenating a long and complex sequence of morphemes. The rules for the formation of the sequence of morphemes are defined according to the language's linguistics phenomena. The following diagram shows the sequence of morphemes for Myanmar verbs. The rules employed by the morphological analyzer and generator play a crucial role in discerning the word structures. Attempting to compile an exhaustive list of all words in a language is impractical. Computational lexicons are typically organized, containing both the stems and affixes of the language, coupled with a representation of morphotactics that delineates how they can coalesce. Numerous approaches exist for modeling morphotactics, with one prevalent method being the utilization of finite state transducers within computational linguistics. [26],[27],[28],[29], [30].

A lexical description is always typed into a text file, which consists of an optional Multichar_Symbols declaration, followed by an optional Declarations section, followed by one or more named LEXICONS, of which exactly one should be named LEXICON Root. We order the morphotactics of each morpheme separately in the lexicon file to show a clear picture of each morpheme combined in the morphological word structure. It is impossible to present the verbal morphotactics of the entire language, which is common to all the varieties of morphemes. We presented general morphotactics that are most relevant and applicable to the whole class of verbs. For example, the finite state transducer network for the morphotactics of aspect maker suffixes is shown below.

```
Multichar_Symbols +N +V +3P +2P
+1P +PL +SG +EXP +CONC +COP +PRI
+COMP +ASP +ADVAL +ADJVAL +NZR
+NZR +SUBOR +REA +PUR +EVD +ACCD
+UNEXP +EXP +JUST +DOBT +HAB +DEAR
+CAUS +MOD +PRB +HON +QWH +INTER
+NEG +CAUS +EUP +SUBJ +IMP +SUG
+PREASP +PAASP +PERASP +PROGASP
+FUTASP +COND +BENF +POL +EMPH
+REAL +CAUS + +GEN +NOM +ACCU +LOC
+ABL +DEMO +Q-Wh +DIST

LEXICON Root
+1P:ငါ Pre_Word_N;   ! pronoun
များပုံစံတူများထည့်ရန်
0:0 Pre_Word_A;
0:0 Suffix_Verb_N;
0:0 Verb_Verb;

!root words end
အရွေ့ဆုံးစာလုံးပြီး----------
----------

LEXICON Pre_Word_N
+NZR+:ဘ Prefix_Verb_N;
0:0 Suffix_Verb_N;

LEXICON Pre_Word_A
+NZR+:ဘ Prefix_Verb_N;
```

Fig. 2: Structure of LEXC file (lexicon file)

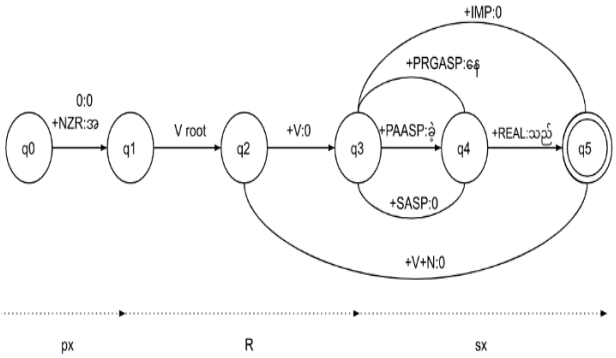

Fig. 3: Diagrammatic representation of FST based on the word formation algorithm for Morphotactics of Myanmar verbs

5.3 Implementation

Developing a computational linguistics tool for morphological analysis and generation in the context of the Myanmar Language involves a combination of technologies, including Xerox's XFST, LEXC tools, FOMA, Python, Java, and other open-source tools. Manual implementation of the language's morphotactics uses the LEXC formalism. The FOMA toolset compiles this lexicon into a finite state transducer capable of analyzing and generating words' surface and lexical forms. Additionally, Integrated Development Environments (IDEs) like Python and Java are employed to create a user-friendly graphical interface.

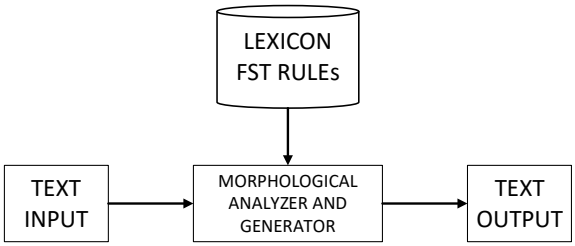

Fig. 4: Fundamental design of a system

```
3828948  တင်သွင်း+V+NZR+PL+BENF+CAUS+POL တင်သွင်းခြင်းများအတွက်ကြောင့်ပါ
3828949  တင်သွင်း+V+NZR+PL+BENF+CAUS+POL+EMPH  တင်သွင်းခြင်းများအတွက်ကြောင့်ပါပဲ
3828950  တင်သွင်း+V+NZR+PL+BENF+CAUS+POL+EMPHREAL  တင်သွင်းခြင်းများအတွက်ကြောင့်ပါပဲတဲ့
3828951  တင်ဆက်+V+NZR  တင်ဆက်ခြင်း
3828952  တင်ဆက်+V+NZR+SG  တင်ဆက်ခြင်း
3828953  တင်ဆက်+V+NZR+SG+ABL  တင်ဆက်ခြင်းထံမှ
3828954  တင်ဆက်+V+NZR+SG+LOC  တင်ဆက်ခြင်း၌
3828955  တင်ဆက်+V+NZR+SG+ACCU  တင်ဆက်ခြင်းကို
3828956  တင်ဆက်+V+NZR+SG+NOM  တင်ဆက်ခြင်းက
3828957  တင်ဆက်+V+NZR+SG+POL  တင်ဆက်ခြင်းပါ
3828958  တင်ဆက်+V+NZR+SG+POL+EMPH  တင်ဆက်ခြင်းပါပဲ
3828959  တင်ဆက်+V+NZR+SG+POL+EMPHREAL  တင်ဆက်ခြင်းပါပဲတဲ့
3828960  တင်ဆက်+V+NZR+SG+GEN  တင်ဆက်ခြင်း၏
3828961  တင်ဆက်+V+NZR+SG+GEN+MAT  တင်ဆက်ခြင်း၏အကြောင်း
3828962  တင်ဆက်+V+NZR+SG+GEN+MAT+POL  တင်ဆက်ခြင်း၏အကြောင်းပါ
3828963  တင်ဆက်+V+NZR+SG+GEN+MAT+POL+EMPH  တင်ဆက်ခြင်း၏အကြောင်းပါပဲ
3828964  တင်ဆက်+V+NZR+SG+GEN+MAT+POL+EMPHREAL  တင်ဆက်ခြင်း၏အကြောင်းပါပဲတဲ့
3828965  တင်ဆက်+V+NZR+SG+GEN  တင်ဆက်ခြင်း
3828966  တင်ဆက်+V+NZR+SG+BENF  တင်ဆက်ခြင်းအတွက်
3828967  တင်ဆက်+V+NZR+SG+BENF+CAUS တင်ဆက်ခြင်းအတွက်ကြောင့်
3828968  တင်ဆက်+V+NZR+SG+BENF+CAUS+POL တင်ဆက်ခြင်းအတွက်ကြောင့်ပါ
3828969  တင်ဆက်+V+NZR+SG+BENF+CAUS+POL+EMPH  တင်ဆက်ခြင်းအတွက်ကြောင့်ပါပဲ
3828970  တင်ဆက်+V+NZR+SG+BENF+CAUS+POL+EMPHREAL  တင်ဆက်ခြင်းအတွက်ကြောင့်ပါပဲတဲ့
3828971  တင်ဆက်+V+NZR+PL  တင်ဆက်ခြင်းများ
3828972  တင်ဆက်+V+NZR+PL+ABL  တင်ဆက်ခြင်းများထံမှ
3828973  တင်ဆက်+V+NZR+PL+LOC  တင်ဆက်ခြင်းများ၌
3828974  တင်ဆက်+V+NZR+PL+ACCU  တင်ဆက်ခြင်းများကို
```

Fig 5: Output of a system

In our previous paper, we presented the development of MM-Morph: a computational linguistic tool for morphological analysis and generation of Myanmar nouns []. In this paper, we integrated the MAG of Myanmar verbs into MM-Morph to improve the system's capabilities. The system can also be integrated into other NLP tasks like spell checking, word suggestion, machine translation, etc.

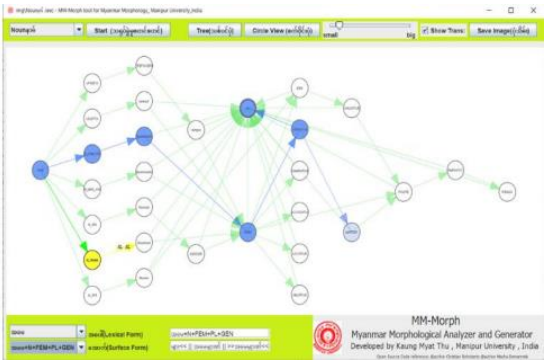

Fig. 6: GUI of MM-Morph tool for morphological analyzer and generator of Myanmar nouns.

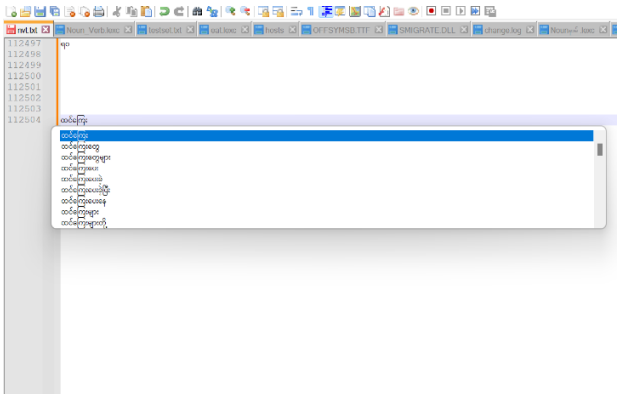

Fig. 7: Example of the use of the output of the system in word suggestion or spell correction

6. Evaluation

We have conducted a small evaluation of our finite state language model to determine the efficiency, sufficiency of the technique, and coverage of various word forms of the verb class of the Myanmar language. To evaluate the system, we collected 100 root tokens of process and action verbs for verbal categories from the above sources. The percentage analysis against the number of entries in the selected verbal source lexica is shown in the following tables.

Table 3: Statistics of the lexical entry and total word forms per entry

Verbal Category				
Root type	No. of the lexical entry	No. of word forms generated per entry	Total word forms	Wrong analysis per entry (Average)
Process	30	63	1890	12
Action	70	68	4760	15

Table 4: Analysis of verbal category word forms

Verbal Category				
Root type	No. of word forms generated per entry	wrong analysis per entry (Average)	% of wrong analysis	% of correct analysis
Process	63	12	19%	81%
Action	68	15	22%	78%

An evaluation of the results yielded by the finite state model has illuminated certain areas where the model encountered challenges. Primarily, the model exhibited deficiencies in handling discrepancies in spelling within word forms and inappropriately combining morphemes, likely attributed to the absence of well-defined grammar rules. Since we have not taken into account instances of reduplication and compounding, our model does not encompass the phenomenon of word compounding. (e.g., "တောနက်" / deep jungle, "ချစ်သက်သေ" / the proof of love) and reduplication of words (e.g.," ချစ်ချစ်ခင်ခင်," "လှလှပပ").

7. Conclusion and Future Work

In this paper, we studied the linguistic phenomena of Myanmar verbs by classifying their features. Being a low-resource language, we have spent too much time studying the linguistics phenomena of the Myanmar language. A language possessing diminished computational underpinning engenders a substantial impediment within the digital epoch, with the Burmese language constituting no exemption. There is still no publicly accessible morphological analyzer or generator for the Myanmar language. As for the Myanmar Language, there aren't many study reports or developments for Morphological analysis and generation. Therefore, we have designed MM-Morph tools to encourage research on the Myanmar language. This paper is presented as part of the research titled "Morphological Analysis and Generation for Myanmar Language with Finite State Techniques. MM-Morph is the first known morphological analyzer and generator for the Myanmar language based on its linguistic features.

Morphological analysis and generation are primarily concerned with the linguistic aspects of a language. This research presents a significant challenge due to the limited availability of linguistic resources for the Myanmar language. Extensive efforts have been dedicated to exploring the distinctive linguistic features inherent in the Myanmar language. Additionally, ongoing endeavors encompass the incorporation of Adverbs and Adjectives into the MM-Morph tool. This holistic approach aims to create a comprehensive morphological analyzer and generator capable of effectively addressing intricate linguistic phenomena.

References

[1] Goldsmith, John. (2001). Unsupervised learning of the morphology of a natural language. Computational linguistics, 27(2):153–198, 2001.

[2] Goldsmith, John. (2006). An Algorithm for the Unsupervised Learning of Morphology. Natural Language Engineering. 12. 353-371. 10.1017/S1351324905004055.

[3] Welgama, Viraj & Weerasinghe, Ruvan & Niranjan, Mahesan. (2013). Evaluating a Machine Learning Approach to Sinhala Morphological Analysis. 10th International Conference on Natural Language Processing, At: Noida, India

[4] John Lee. (2008). A Nearest-Neighbour Approach to the Automatic Analysis of Ancient Greek Morphology. CoNLL 2008: Proceedings of the 12th Conference on Computational Natural Language Learning, Manchester, August 2008: 127–134

[5] Anand Kumar, M. & Dhanalakshmi, V. & Kp, Soman & Sankaravelayuthan, Rajendran. (2010). A Sequence Labeling Approach to Morphological Analyzer for Tamil Language. (IJCSE) International Journal on Computer Science and Engineering Volume 02, Issue No. 06, Page no (2201-2208),2010.

[6] Canasai Kruengkrai, Virach Sornlertlamvanich, Hitoshi Isahara. (2006). A Conditional Random Field Framework for Thai Morphological Analysis. In Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC-06), May 24-26, 2006. Genoa, Italy.)

[7] P. J. Antony, Dr. M. Anand Kumar, and Dr. Soman K. P. (2006). Paradigm based morphological analyzer for Kannada language using machine learning approach. International journal on-Advances in Computer Science and Technology (ACST), ISSN 0973-6107, vol. 3, pp. 457–481.

[8] Shah D.N., Bhadka H. (2020). Paradigm-Based Morphological Analyzer for the Gujarati Language. Advances in Intelligent Systems and Computing, vol 989. Springer, Singapore. https://doi.org/10.1007/978-981-13-8618-3_50

[9] Jedrzejowicz, Piotr & Strychowski, Jakub. (2005). A Neural Network Based Morphological Analyser of the Natural Language. 199-208. 10.1007/3-540-32392-9_21.

[10] Premjith, B., Soman, K. P., & Kumar, M. A. (2018). A deep learning approach for Malayalam morphological analysis at character level. Procedia computer science, 132, 47-54.

[11] Kimmo Koskenniemi. (1983). Two-level morphology. Ph.D. thesis, University of Helsinki.

[12] Lauri Karttunen and Kenneth R Beesley. (2012). A short history of two-level morphology. ESSLLI-2001 Special Event titled" Twenty Years of Finite-State Morphology.

[13] Katushemererwe, F., & Hanneforth, T. (2010). Finite State Methods in Morphological Analysis of Runyakitara Verbs. Nordic Journal of African Studies, 19, 22-22.

[14] Sarveswaran, K., Dias, G., & Butt, M. (2018). ThamizhiFST: A Morphological Analyser and Generator for Tamil Verbs. 2018 3rd International Conference on Information Technology Research (ICITR), 1-6.

[15] Kengatharaiyer, Sarveswaran & Dias, Gihan & Butt, Miriam. (2019). Using Meta-Morph Rules to develop Morphological Analysers: A case study concerning Tamil. 10.18653/v1/W19-3111.

[16] Kayabaş, A., Schmid, H., Topcu, A., & Kiliç, Ö. (2019). TRMOR: a finite-state-based morphological analyzer for Turkish. Turkish Journal of Electrical Engineering and Computer Sciences, 27, 3837-3851.

[17] Kenneth R Beesley and Lauri Karttunen. (2003). Finite-state morphology: Xerox tools and techniques. CSLI Publications, Stanford.

[18] Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. (2007). OpenFst: A general and efficient weighted finite state transducer library. In International Conference on Implementation and Application of Automata, pages 11–23. Springer.

[19] Krister Lindén, Miikka Silfverberg, and Tommi Pirinen. (2009). HFST tools for morphology–an efficient open-source package for construction of morphological analyzers. In International Workshop on Systems and Frameworks for Computational Morphology, pages 28–47. Springer.

[20] Mans Hulden. (2009). Foma: a finite-state compiler and library. In Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Demonstrations Session, pages 29–32. Association for Computational Linguistics.

[21] Abebe, Tewodros & Washington, Jonathan & Gasser, Michael & Yimam, Baye. (2018). A Finite-State Morphological Analyzer for Wolaytta. Information

and Communication Technology for Development for Africa.10.1007/978-3-319-95153-9_2.

[22] Rahman, Mirzanur & Sarma, Shikhar. (2015). An Implementation of Apertium Based Assamese Morphological Analyzer. International Journal on Natural Language Computing. 4. 10.5121/ijnlc.2015.4102.

[23] Zueva, A., Kuznetsova, A., & Tyers, F. (2020). A Finite-State Morphological Analyser for Evenki. Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 2581–2589.

[24] Ammari, Rachid & Zenkoua, Ahbib. (2021). APMorph: finite-state transducer for Amazigh pronominal morphology. International Journal of Electrical and Computer Engineering (IJECE). 11. 699. 10.11591/ijece.v11i1.pp699-706.

[25] Keleg, A., Tyers, F.M., Howell, N., & Pirinen, T.A. (2020). An Unsupervised Method for Weighting Finite-state Morphological Analyzers. Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 3842–3850

[26] Phyue, S. L., & Thida, A. (2012). Morphological Processor for Inflectional Case of Multipurpose Lexico-Conceptual KnowledgeResource. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 1(7), 157-163.

[27] Latt, T. M., & Thida, A. (2018, June). An Analysis of Myanmar Inflectional Morphology Using Finite-state Method. In 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS) (pp. 297-302). IEEE.

[28] Sgarbas. (2000). A Straight Forward Approach to Morphological Analysis and Synthesis, Sgarbas (et. all), Proceeding of COMLEX 2000, Greece.

[29] Sarmah, J., Sarma, S.K., & Barman, A. (2019). Development of Assamese Rule-based Stemmer using WordNet. GWC.

[30] Debbarma, K., Patra, B. G., Das, D., & Bandyopadhyay, S. (2012, December). Morphological Analyzer for Kokborok. In Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing (pp. 41-52).

[31] Huxley, A. (2003). The Burmese Script. Journal of the Royal Asiatic Society, 133(2), 197-207.

[32] Luce, G. H. (1985). Early Inscriptions of Burma. In Indic Scripts of Southeast Asia (pp. 45-63). Cornell University Press.

[33] Okell, J. (2001). The Early Mon Tradition. In Myanmar: State, Society and Ethnicity (pp. 39-57). Institute of Southeast Asian Studies.

[34] Juneja, V., Singh, S., Jain, V., Pandey, K.K., Dhabliya, D., Gupta, A., Pandey, D. Optimization-based data science for an IoT service applicable in smart cities (2023) Handbook of Research on Data-Driven Mathematical Modeling in Smart Cities, pp. 300-321.

[35] Dhabliya, D. Security analysis of password schemes using virtual environment (2019) International Journal of Advanced Science and Technology, 28 (20), pp. 1334-1339