Khmer Treebank Construction via Interactive Tree Visualization

Bonpagna Kann[1], Thodsaporn Chay-intr[2], Thanaruk Theeramunkong[2], Hour Kaing[1]

***Abstract*— Despite the fact that there are a number of researches working on Khmer Language in the field of Natural Language Processing along with some resources regarding words segmentation and POS Tagging, we still lack of high-level resources regarding syntax, Treebanks and grammars, for example. This paper illustrates the semi-automatic framework of constructing Khmer Treebank and the extraction of the Khmer grammar rules from a set of sentences taken from the Khmer grammar books. Initially, these sentences will be manually annotated and processed to generate a number of grammar rules with their probabilities once the Treebank is obtained. In our experiments, the annotated trees and the extracted grammar rules are analyzed in both quantitative and qualitative way. Finally, the results will be evaluated in three evaluation processes including Self-Consistency, 5-Fold Cross-Validation, Leave-One-Out Cross-Validation along with the three validation methods such as Precision, Recall, F1-Measure. According to the result of the three validations, Self-Consistency has shown the best result with more than 92%, followed by the Leave-One-Out Cross-Validation and 5-Fold Cross Validation with the average of 88% and 75% respectively. On the other hand, the crossing bracket data shows that Leave-One-Out Cross Validation holds the highest average with 96% while the other two are 85% and 89%, respectively.**

***Keywords* — Treebank Construction, Grammar Construction, Visualization Tool, Syntactic Parsing.**

I. Introduction

Treebank is considered as the essential resource in the development in the comprehension of a language in Natural language processing (NLP) as it plays a vital role as the annotated resources for the research and development of the language. For example, The Penn Treebank [1], one of popular sources of the annotated text corpus widely available within the NLP community, led to the advancement of the first competent English parsers and the breakthrough of the statistical revolution within NLP as it provides the crucial training and testing data for the research process including parser and machine translator.

Currently, there are projects of treebank construction in many languages with the aim to expand the development of the language resources in a variety of languages. According to [2], a project for building Asian Language Treebank (ALT) was launched with the purpose to develop the state-of-the-art Asian NLP technologies through the open collaboration for developing and using ALT which initially developed in seven languages including English, Indonesian, Japanese, Khmer, Malay, Burmese, and Vietnamese. However, Khmer language processing is still in a limited condition as the reason of the lack of high-level syntactic resources which is necessary to build the treebank along with the rich resources of linguistics knowledge as its complexity.

Despite the fact that there are a number of researches working on the word segmentation and POS tagging, there is a lack of study on the Khmer grammar since it is required a lot of linguistics knowledge as its complexity. In addition, a high-level syntactic resources, such as treebank and grammar in Khmer language are very limited as it is needed to do more manual work to obtain the data. This paper will illustrate a framework of constructing Khmer treebank and extraction of grammar rules. We extract grammar rules from the annotated treebank which we manually construct. In the experiments, we select a hundred sentences from Khmer grammar books as the sources for the syntactic annotation and the results will be analyzed. In Section II, we present the basic language structure of Khmer grammar and the previous researches. Next, in Section III, the framework to construct the Khmer treebank, extract and revise Context-Free-Grammar rules. Following that, the materials, results along with the evaluation and error analysis will be depicted in the Section IV. Finally, in Section V, the paper will be concluded and the future work will be described for the future development of Khmer Treebank.

II. Related Works

In order to illustrate the integrated framework and the tree visualization tool, it is necessary to demonstrate the adversity of Khmer language and the research works of the tree bank construction.

A. Khmer Grammar Focus: Language Structure

There is a difficulty in recognizing the structure of the word and sentence due to the reason that there is no capitalized structure and no delimiter between each word or sentence in Khmer language [3]. In addition, couple words can be combined to create one sentence and multiple words can also be merged to create a new word, known as compound words [4]. For example, “អ្នកធ្វើការ” (Meaning: worker, pronunciation: /neak/ /tvəə/ /ka/), which is created by the combination from “អ្នក” (meaning: a person, pronunciation: /neak/), “ធ្វើ” (Meaning: do, pronunciation: /tvəə/) and “ការ” (Meaning: work, pronunciation: /ka/). In Khmer sentence structure, SVO has been defined as the basic structure of the sentence (Subject-Verb-Object) in addition to the head-initial (modified modifier) which is used to modified word order, and the noun classifier system [5]. Additionally, there are also

[1] *Institute of Technology of Cambodia, Phnom Penh, Cambodia (bonpagnakann@gmail.com, kainghour@gmail.com)*
[2] *Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani, Thailand (t.chayintr@icloud.com, thanaruk@siit.tu.ac.th)*

many other structures beside SVO according to the context including OSV (object-subject-verb), or SV(subject-verb). Due to the complexity and variety of the language structure of Khmer language, it is a big challenge for natural language processing (NLP) or computational linguistics to identify and analyze the ambiguity of the language.

B. The Previous Researches of Khmer NLP Resources

In a previous Khmer lexicon research works on word segmentation, it is denoted that the maximum matching method for Khmer word segmentation is based on the publicly available Khmer dictionary. Moreover, the Conditional Random Fields (CRFs) has been used as an approach to use in word segmentation of Khmer language [6]. Moreover, the Public POS tagged data is only found at PAN localization's website [7]. The corpus is about 3,000 manually annotated sentences, which were used to train for POS tagger for Khmer language [8]. The tool is published by PAN localization. Along with that, NIPTICT has been manually annotating POS tagged data about 30,000 sentences, which is segmented by Khmer Word Segmenter [6] though the data was not open to the public.

III. KHMER TREEBANK CONSTRUCTION AND GRAMMAR EXTRACTION

It is essential to perform pre-processing tasks in the Treebank annotation. Furthermore, a representative annotation is necessary to display the syntactic trees in Treebank repository.

A. Treebank Construction Framework

In the previous study of Thai Treebank, the Treebank with semi-automatic framework has been constructed by using Thai fairy tales as a test-bed [9]. The framework is shown in Fig. 1. According to their work, the framework consists of four main components including sentence boundary annotation, word segmentation, POS tagging and syntactic tree bracketing and labeling. In our framework, word segmentation and part-of-speech tagging can be performed automatically by existing systems, sentence boundaries and syntactic trees are annotated manually.

1) Sentence Boundary Annotation (Seed Text Sentences): According to [10], Khmer language has no explicit sentence boundary marker if we consider the comparison with other languages, including English, French or Japanese. The structure annotation is subjective in most cases, and it varies by individual. Hence, it is needed for the annotators to have the common understanding on Khmer sentence structure. In practical way, we construct the Khmer sentence structure in the similar way using the English sentence structure.

2) Word Segmentation: Similarly, there is also no explicit word boundary delimiters such as blank space to separate between each word. In addition, there is a more complex structure to the form of the word, causing the Khmer Unicode standard ordering of character components to permit different orders that lead to the same visual representation; exactly looking word, but different character order. In the large data scale, it can be a heavy task to segment the word boundary; hence, the technique from [10] which provided the maximum matching technique "Bi-directional Maximal Matching" and the accuracy result of 98.13%.

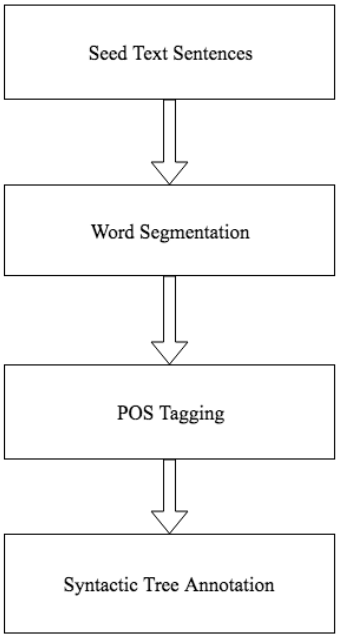

Fig. 1 Pre-processing tasks.

3) POS Tagging: According to the previous work in [7], for example, the POS tagger in Khmer language has been conducted based on Decision Tree model, for semi-automatic tagging of Khmer language. Moreover, as there are many problems regarding the unknown words in Khmer language, another approach in [11] has been suggested that the hybrid approach, a combination model of rule-based and trigram models, plays a vital role in handling the unknown word problem in Khmer part-of-speech tagging by making use of both internal structure of the word and surrounding contextual information to predict the part-of-speech of unknown words.

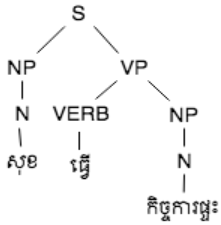

Fig. 2 Example of the syntax tree.

4) Treebank Representations: To provide a better representation of the syntactic trees in the Treebank, [9] has conducted the labelled bracket in text-based approaches to illustrate phrase structures for syntactic tree in Thai Treebank. The syntactic tree label bracketing provides the efficiency in grammar derivation of the language. Hence, we manually built the syntactic labelled brackets for each sentence and analyze its structure in order to extract the grammar. The set of the syntactic trees will be represented in a labelled bracket format

as shown in Table I. As a result, Fig. 2 illustrates the syntactic trees which is transformed from the labelled bracket format. Furthermore, the process of the annotation and correction is performed by one main annotator and another consultant by discussing each sentence to annotate the most likely sentence with POS and syntactic tree.

5) Annotation Tools: According to Eryiğit, ITU Treebank Annotation was a development in Treebank annotation [12]. It has been used to build the dependency tree by steps, guiding the annotation for the minimization of the number of errors made by human in annotation process using grammar and dependency parser. However, Stenetorp et al., 2012 [13] has shown a new web-based text annotation tools, providing annotators features for a better speed and consistency of the tagging process. We have also implemented the visualization tool in this research to annotate the trees. By using the visualization tool, it can facilitate the annotators to construct and verify their work more efficiently and effectively due to the reason that the annotation tools for Khmer language is still limited. The online tool which is used to illustrate the labelled brackets of the sentences into visual images for the annotator is the web-based application from the link, http://mshang.ca/syntree/. It is the Syntax Tree Generator that we can generate the tree from the labelled brackets shown in the Fig. 3.

TABLE I
EXAMPLE OF THE LABELLED BRACKETS

Original Text	សុខធ្វើកិច្ចការផ្ទះ
Translation	Sok does homework.
Labelled bracket	[S [NP [N សុខ]] [VP [VERB ធ្វើ] [NP [N កិច្ចការផ្ទះ]]]]

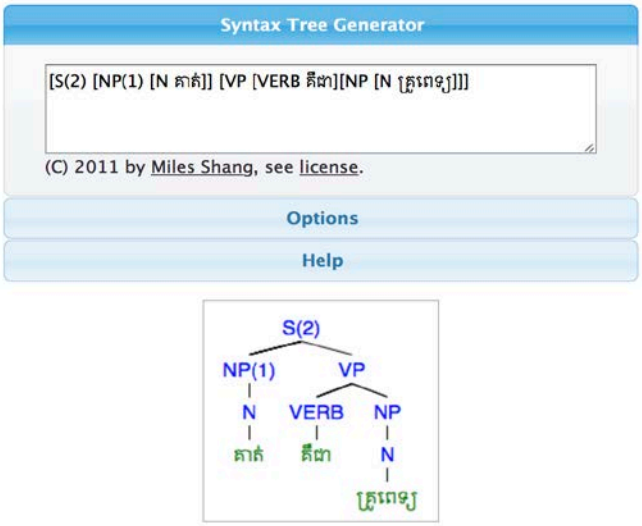

Fig. 3 Tree generated by Syntax Tree Generator.

After the labelled bracket has been created as tree, each of them is added to a text file shown in Fig. 4. After a number of trees have been added to the file, we can obtain the list of trees which is called treebank, illustrated as following.

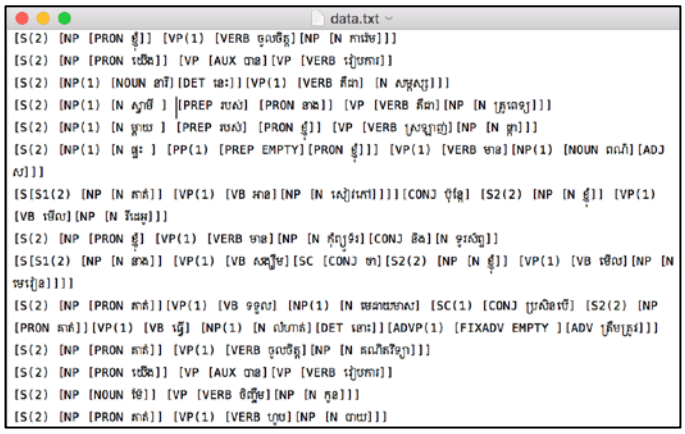

Fig. 4 The text file containing labelled-bracket sentences.

B. Grammar of Khmer Language

As the context-free grammar rules are implemented, there is the possibility that ambiguity can occur due to the variousness of structures to depict the language. According to [14], there is the possibility of having more than one method to parse the string into parse trees which means that one string can generate multiple parse trees. On the other hand, by using the characteristics and generalization, we can derive a large number of sentences in the language [15]. Hence, the conciseness and the semantic encapsulation for the Khmer Treebank construction are analyzed to extract the grammar of Khmer language.

The 100 sentences in Khmer consist of three main clauses, including simple sentences, compound sentences and complex sentences. Simple Sentence is a sentence that contains only one clause and that clause is independent clause is called simple sentence [16] For example, ខ្ញុំចូលចិត្តការរៀម (I like ice-cream). In this sentence, there is only one independent clause in the SVO structure. The syntactic structure of this sentence is illustrated in the Fig. 5.

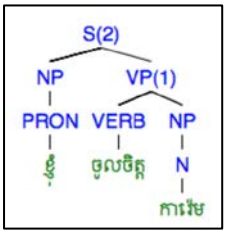

Fig. 5 The syntactic tree of simple sentence.

Compound Sentence is a sentence that contains two or more independent clauses joined by coordinate conjunctions [16]. In Khmer language, the coordinate conjunctions include “និង” (And), “សម្រាប់” (For), “ប៉ុន្តែ” (But), “ឬ” (Or), “តែ” (But), “ចំណែក” (And), etc [17]. Fig. 6 is an example of the syntactic tree of the compound sentence.

Complex Sentence is a sentence that contains at least one dependent clause with independent clause connected by the subordinate conjunctions [16]. In Khmer language, the coordinate conjunctions include “បន្ទាប់ពី” (After), “ប្រសិនបើ” (If), “ព្រោះ”(Because), “ដែល” (Which), “ថ្វីត្បិតតែ” (Even

though), "ទោះបី" (Although), etc [17]. Here is an example of the syntactic tree of the complex sentence illustrated in Fig. 7.

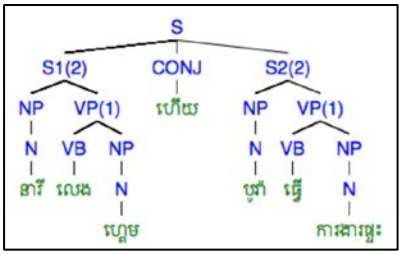

Fig. 6 The syntactic tree of compound sentence.

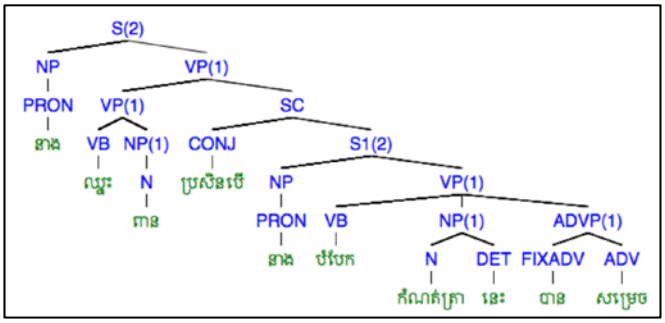

Fig. 7 The syntactic tree of complex sentence.

After the grammar rules are extracted from the Treebank, the parser is implemented to parse the sentences and evaluate the number of generated trees. For the efficient result, we minimize the number of generated trees while retaining their structure. By doing this, the grammar rules are adjusted reiteratively as we eliminate the unnecessary rules.

IV. Experiments

A. Materials

In the experiment of this research, a set of sentences has been selected from the Khmer grammar book as the material as there are clear and common structures when can be easily used for the probabilistic grammar extraction and iterative refinement in the training process. After the sentences have been processed and added to the Treebank manually, 100 syntactic trees were generated and 68 probabilistic grammar rules have been extracted. Characteristic of our materials are shown in Table II.

Table II
Description of the Materials

#Sentences	#Word/Sentence			#Character/Word		
	Min.	Max.	Avg.	Min.	Max.	Avg.
100	2	16	5.62	1	13	4.72
	Total		562	Total		2654

B. Results and Discussion

The results obtained from building the Khmer Treebank and extraction of the Khmer grammar rules based on the sentences from the grammar books will be presented in this part. There are two main processes in the framework of the construction of Khmer treebank including word segmentation and POS tagging using existing tools. However, the boundary annotation of the executed sentences and syntactic tree bracketing/labelling are done manually.

According to the results of the processes, we constructed 100 syntactic trees, 45 POSs, 12 phrase tagsets as illustrated in Appendix A and 65 CFG rules are straightforwardly extracted from the Treebank. The extracted Khmer grammar for sentences with its frequency are shown in Table III and all grammar rules, without duplicated occurrences in the same sentences, are shown in Appendix B.

Table III
Data of Khmer Context-Free Grammar (CFG)

Grammar	Frequency	Probability
NP -> N	105	0.44872
S -> NP VP	86	0.86869
VP -> VERB NP	71	0.50000
NP -> PRON	53	0.2265
NP -> N PP	29	0.12393
PP -> PREP PRON	22	0.51163
S1 -> NP VP	17	1
PP -> PREP NP	17	0.39535
ADVP -> ADV	14	0.58333
VP -> AUX VP	13	0.09155
ADJP -> ADJ	12	0.04762
S2 -> NP VP	12	0.85714
VP -> VERB ADVP	10	0.03521

C. Evaluation and Error Analysis

In the evaluation process, there are three evaluation methods including Self-Consistency (SC), 5-Fold Cross Validation (5-Fold CV), and Leave-One-Out Cross-Validation (LOO CV) to test the 100 syntactic tree of the sentences. As processing the raw textual sentences might not be effective to evaluate the sentences, we decided to use the POS-Tagged sentence to submit in order to avoid the fragments of the word segmentation and result of the POS tagging. Furthermore, we have implemented three measurements to ensure the preciseness of the grammar extracted from the sentences. The measurements include PARSEVAL measurements, which are labeled precision, Recall, F-Measure with their standard deviation, and cross bracketing which are performed by manipulating from the successful grammar extraction.

Table IV
Parsable Ratio and Cross Bracket

Validation	Parsable Ratio	# CB(s)/Sentence
	Average ± (SD)	Average ± (SD)
SC	0.99 ± (0.01)	0.89 ± (1.51)
5-fold CV	0.82 ± (0.04)	0.85 ± (1.31)
LOO CV	0.96 ± (0.04)	0.96 ± (1.45)

(SC: Self-Consistency, CV: Cross-Validation, LOO: Leave-One-Out, CB: Cross Bracket)

According to the results of our experiments in Table IV, the program can parsed the POS-tagged sentences using the grammars extracted from the sentences by the average ratio of 0.82 in 5-Fold Cross Validation while performing almost perfectly in Self-Consistency and Leave-One-Out Cross

Validation with the average ratio of 0.99 and 0.96 respectively. In addition, despite the difference of the 5-Fold Cross Validation from the other two validation methods, these average number of cross-brackets per sentence at around 0.90 has shown a positive result of the extracted grammar. Considering the result based on Table V, there are also significant outcomes obtained. In all three measurements, the Self-Consistency got the highest average of approximately 93% average, followed by the Leave-One-Out Cross-Validation, 88%. However, there is a lower average of the 5-Fold Cross Validation which stays at only around 75%, more than 10% lower compared to the other two.

TABLE V
PRECISION, RECALL, F1-MEASURE

Validation	Precision	Recall	F1-Measure
	Average ± (SD)	Average ± (SD)	Average ± (SD)
SC	93.13 ± (12.89)	92.79 ± (12.65)	92.79 ± (12.27)
5-Fold CV	74.88 ± (19.37)	76.15 ± (19.30)	75.21 ± (18.73)
LOO CV	87.43 ± (13.78)	88.38 ± (13.46)	87.64 ± (12.90)

(SC: Self-Consistency, CV: Cross-Validation, LOO: Leave-One-Out, CB: Cross Bracket)

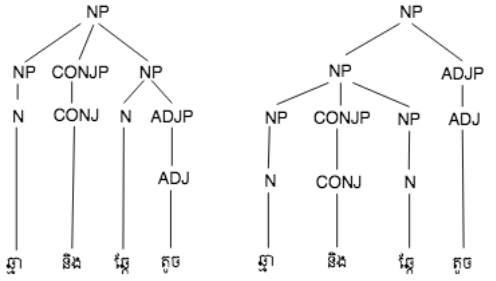

Fig. 8 Examples of syntactic ambiguity.

While Khmer Treebank has been being constructed, the problems and difficulties were also arisen, one of which was the lexical ambiguity and syntactic ambiguity which includes conjunction, lexical, sentence structure and clause ambiguity. In order to avoid these problems, the syntactic trees were manually annotated with the awareness of ambiguity to improve the quality of the syntactic trees.

1) Lexical Ambiguity: Some words have multiple meaning according to the context in the sentence, which causes the ambiguity in the lexical way. For example, the word "កន្សែង" can be verb or noun depending on its context in the sentence, such as (a) ស្តេចកន្សែង (the king cries), (b) យកកន្សែងប្រើ (Take the towel to use). Hence, it is needed to annotate the accurate word class to obtain the correct syntactic tree.

2) Syntactic Ambiguity: It is important to consider the semantics during annotating each phrase as the reason of the manifestation of the sentence illustrated by the hierarchy of the phrases in the syntactic tree. According to the Fig. 8, an example has been demonstrated regarding syntactic ambiguity as there are two ways to annotate "ឆ្មានិងឆ្កែធំ" (cat and big dog) including: (a) "ឆ្មានិងឆ្កែធំ" (cat and big dog) and (b) ឆ្មាធំនិងឆ្កែធំ (big cat and big dog). Hence, it is important it is needed to consider its context and sematic meaning in order to avoid this ambiguity.

V. CONCLUSION AND FUTURE WORKS

Despite the fact that there are a number of researches working on the word segmentation and POS tagging of Khmer language, there is a lack of the syntactic resources including Khmer grammar and Treebank. However, those resources play a vital part in the research of the grammar as they can be used as the background resources to develop the grammar structure from the sentences. As it is essential to work intensively on NLP regarding the characteristics of the language as well as its syntactic and semantic structure in order to build Khmer Treebank, this paper is an initial step of constructing the grammars of Khmer language through Khmer Treebank using the sentences from the Khmer grammar books to enhance the quality of the Khmer language resources. After we have constructed the Treebank, we can extract the grammar rules from Khmer treebank.

In our experiments, the syntactic trees as well as the grammar rules are processed and analyzed in both quantitative and qualitative ways. The small set of sentences are manually converted to annotated syntactic trees represented by the labelled brackets before the set of initial grammar rules with their probabilities are derived. Then, another set of experiments are conducted to illustrate the frequency of the extracted grammar rules. Next, the set of extracted grammar rules are applied for the evaluation process, estimating PARSEVAL Measurements of a set of successful syntactic tree in three types of validation including Self-Consistency, 5-Fold Cross Validation, and Leave-One-Out Cross-Validation.

According to the result of the three validations, it is noticeable that the Self-Consistency has shown the best result with more than 92%, succeeded by the Leave-One-Out Cross-Validation and 5-Fold Cross Validation with the average of 88% and 75% respectively. However, the crossing bracket data shows that Leave-One-Out Cross Validation holds the highest average with 0.96 while the other two remain around 0.85 and 0.89. Finally, we also discuss the difficulties and error analysis occurred during the construction of Khmer Treebank including the lexical and syntactic ambiguity. In the future, we plan to increase the size of training data by providing more syntactic trees for Khmer Treebank using manual and statistic annotation, adding and varying more structures of the sentence to generate more grammar rules retaining the syntactic and semantic structure to refine the statistic of correctness of our training Treebank. On top of that, we hope to continually develop the application of grammar suggestion using the Treebank and the grammar rules as the root sources to develop the platform to fulfil the demand of a domain-specific Treebank for various research and applications.

ACKNOWLEDGMENT

This work is conducted with the cooperation of Sirindhorn International Institute of Technology, Thammasat University and Institute of Technology of Cambodia. We would like to thank the reviewers for their valuable comments and suggestions to improve the quality of our paper.

REFERENCES

[1] M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz, "Building a Large Annotated Corpus of English: The Penn Treebank," *Comput. Linguist.*, Vol. 19, No. 2, pp. 313–330, 1993.

[2] H. Riza, M. Purwoadi, T. Uliniansyah, and B. Pengkajian, "Introduction of the Asian Language Treebank," *Proceedings of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Technique (O-COCOSDA)*, 2016, pp. 26–28.

[3] C. Nou and W. Kameyama, "A Rule-Based Proper Noun Recognition for Khmer Part-of-Speech Tagger," *The 69th National Convention of IPSJ*, 2007, pp. 2–385-586.

[4] O. Boonyarith, "Derivatives in Khmer Compound Words," *Mon-Khmer Stud. 38 A J. Southeast Asian Lang. Cult. Spec. Vol. Dedic. to Dr. David Thomas*, 2009, pp. 173–183.

[5] S. Prasomsuk and P. Mol, "Thai to Khmer Rule-Based Machine Translation using Reordering Word to Phrase," International Journal of Computer Theory and Engineering, Vol. 9, No. 3, pp. 223-228, 2017.

[6] V. Chea, Y.K. Thu, C. Ding, M. Utiyama, A. Finch, and E. Sumita, "Khmer Word Segmentation Using Conditional Random Fields," *Khmer Natural Language Processing*, 2015, pp. 1-8.

[7] IDRC, "Khmer Part-of-Speech Tagger," PAN Localization Cambodia (PLC) of IDRC, Project Report, pp. 1-10, 2008.

[8] C. Nou and W. Kameyama, "Khmer POS Tagger: A Transformation-Based Approach with Hybrid Unknown Word Handling," *ICSC 2007 Int. Conf. Semant. Comput.*, 2007, pp. 482–489.

[9] T. Chay-intr and T. Theeramunkong, "Towards Thai Treebank Construction and Grammar Derivation," *Proc. Jt. Int. Symp. Artif. Intell. Nat. Lang. Process. (iSAI-NLP 2017)*, 2017, pp. 133-141.

[10] N. Bi and N. Taing. "Khmer Word Segmentation Based on Bi-directional Maximal Matching for Plaintext and Microsoft Word Document," *2014 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. APSIPA 2014*, 2014, pp. 1-9.

[11] C. Nou and W. Kameyama, "Hybrid Approach for Khmer Unknown Word POS Guessing," *2007 IEEE International Conference on Information Reuse and Integration, IEEE IRI-2007*, 2007, pp. 215–220.

[12] G. Eryiğit, "ITU Treebank Annotation Tool," *Proceedings of the Linguistic Annotation Workshop (LAW '07)*, 2007, pp. 117-120.

[13] P. Stenetorp, S. Pyysalo, G. Topi, T. Ohta, S. Ananiadou, and J. Tsujii, "BRAT : A Web-based Tool for NLP-Assisted Text Annotation," *EACL '12 Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, 2012, pp. 102–107.

[14] C. Brabrand, R. Giegerich, and A. Møller, "Analyzing Ambiguity of Context-free Grammars," *Sci. Comput. Program.*, Vol. 75, No. 3, pp. 176–191, 2010.

[15] T.H. Chen, C.-H. Tseng, and C.-P. Chen, "Automatic Learning of Context-Free Grammar," *Proc. 18th Conf. Comput. Linguist. Speech Process.*, 2006, pp.53–62.

[16] Chandni, R. Narula, and S.K. Sharma, "Identification and Separation of Simple, Compound and Complex Sentences in Punjabi Language," International Journal of Computer Applications & Information Technology, Vol. 6, No. 2, pp. 123–128, 2014.

[17] C. Chhorn, *Khmer Grammar for Students in General*, Phnom Penh, Cambodia, 2002, pp.10-12.

Appendix A.1: Part of Speech Tagsets

No	POS	Description	Grouped Version	Example
1	NCMN	Common noun	N	ឡាន (Car), សកម្មភាព (Action), ខ្មៅដៃ (Pencil)
2	NPRP	Proper noun		សុខា (Sokha) ភ្នំពេញ (Phnom Penh), ថ្ងៃអង្គារ (Tuesday)
3	NCNM	Cardinal number		បី (three), ១០០ (100), ចតុ (four)
4	NONM	Ordinal number		ទីបី (Third), ទីមួយរយ (100th), ទីបួន (fourth)
5	NLBL	Label noun		1, 2, 3, 4, ក, ខ, a, b
6	NTTL	Title noun		លោក (Mr.), លោកស្រី (Mrs.), កញ្ញា (Ms.)
7	PPRS	Personal pronoun		គាត់ (He or she), នាង (She), វា (It)
8	DDAN	Definite determiner, after noun without classifier in between	DET	នេះ (This), ទាំងនេះ (These), ទាំងនោះ (Those)
9	DDAC	Definite determiner, allowing classifier in between		នេះ (This) ទាំងនេះ (These), ទាំងនោះ (Those)
10	DDBQ	Definite determiner, between noun and classifier or preceding quantitative expression		ទាំង (Including), ទៀត (More), គ្រាន់តែ (Just, Only)
11	DDAQ	Definite determiner, following quantitative expression		គត់ (Absolute)
12	DIAC	Indefinite determiner, following noun; allowing classifier in between		មួយណា (Which one), នានា (A variety of)
13	DIBQ	Indefinite determiner, between noun and classifier or preceding quantitative expression		ប្រមាណ (Approximately), ប្រហែល (Around)
14	DIAQ	Indefinite determiner, following quantitative expression		សល់ (Remaining), ខ្វះ (More needed), លើស (Exceeding the needs)
15	DCNM	Determiner, cardinal number expression		ពីរដើម (Pen) ផ្ទះពាក្ខង (House)
16	DONM	Determiner, ordinal number expression		ទីបី (third), ទីមួយរយ (100th), ទីបួន (fourth)
17	PDMN	Demonstrative pronoun		នេះ (This), នោះ (That), ទាំងនេះ (These)
18	PNTR	Interrogative pronoun		តើ (What), ហេតុអ្វី(Why)

No	POS	Description	Grouped Version	Example
19	PREL	Relative pronoun		ដែល (That, Which, Where, Who)
20	VATT	Atrributive verb	VB	ស្អាត (Beautiful), ធំ (Big), ល្អ (Good)
21	VTRA	Transitive verb		សរសេរ, (Write), ទាត់ (Kick) សម្អាត (Clean)
22	VINT	Intransitive verb		ដេក (Sleep), អង្គុយ (Sit), ស្លាប់ (Die)
23	XVBB	Pre-verb auxiliary, in imperative mood		សូម (Please), មេត្តា (Please), កុំ (Do not)
24	XVAE	Post-verb auxiliary		ទៅ, ឡើង, មក
25	ADVN	Adverb with normal form	ADV	លឿន (quickly) ណាស់ (very), ខ្លាំងណាស់ (strongly)
26	ADVI	Adverb with iterative form		គ្រប់ៗគ្នា (every)
27	ADVP	Adverb with prefixed form		យ៉ាង (very)
28	ADVS	Sentential adverb		និយាយអោយខ្លីទៅ (In short), តាមពិត (In fact), និយាយសរុបទៅ (On the whole)
29	CNIT	Unit classifier		ដើម (tall tree) ដប (Bottle), ក្បាល(Animal)
30	CLTV	Collective classifier	CLAS	រ៉ូង (crowd of animal), ក្រុម(Teams), ជាច់(Chopsticks)
31	CMTR	Measurement classifier		ម៉ែត្រ (Meter), លីត្រ (Liter), គីឡូ (Kilometer)
32	CFQC	Frequency classifier		ដង (Times), ជុំ (Rounds)
33	CVBL	Verbal classifier		បង្ខេច (Package), ចំណែក(Pieces), បំណែក (Pieces)
34	JCRG	Coordinating conjunction	CONJ	និង (And), សម្រាប់ (For), ប៉ុន្តែ (But)
35	JCMP	Comparative conjunction		ដូចគ្នាដែល (In the same way), ប្រហាក់ប្រហែលគ្នាដែល (Similarly)
36	JSBR	Subordinating conjunction		ពីព្រោះ (Because), ថ្វីត្បិតតែ (Although), ទោះបី (Even though)
37	RPRE	Preposition		ចំពោះ (to), ដល់ (to), ដើម្បី (in order to)
38	INT	Interjection		អូ (Oh!), ម៉ៅ (Meow), អ៊ុះ (uh)
39	FIXN	Nominal prefix		ការអភិវឌ្ឍន៍ (what), ភាពស្រស់ប្រពៃ(Why)
40	FIXV	Adverbial prefix		យ៉ាង
41	EAFF	Ending for affirmative sentence		ណា, ហ្នឹងណា
42	EITT	Ending for Interrogative sentence		ឬទេ? ឬ? ទេ?
43	ENEG	Ending for Negative sentence		ឡើយ, ទេ
43	NEG	Negator		មិន, អត់, មិនមែន
44	PUNC	Punctuation		+, -, *, /, ៖
45	COMP	Complimentizer		ថា

Appendix A.2: Phrase Tagsets

No	Phrase	Description	Example of Grammar	Example
1	ADJP	Adjective Phrase	FIXADJ + ADJP, ADJP + CLAS	យ៉ាង + ធំ សៀវភៅក្រាស់ + មួយក្បាល
2	ADVP	Adverb Phrase	FIXADJ + ADJP	យ៉ាង +លឿន (Quickly) ភ្លាមនោះ (Immediately)
3	CONJP	Conjunction Phrase	CONJP + When, CONJ	ដូច្នេះ + នៅពេល (Hence + When) ហើយ (And)
4	NP	Noun Phrase	NP + ADJP, NP + DET	មនុស្ស + ល្អគ្រប់គ្រាន់ (Person + Good Enough) មនុស្ស + ម្នាក់ (One Person)
5	PP	Prepositional Phrase	PREP + NP PREP + NP + RP	របស់ + នាង (Of + her) សម្រាប់ + ខ្ញុំ + ចុះ (For me)
6	S	Sentence	NP + VP NP + VP + PP	ខ្ញុំ + ជីកទឹក (I drink water) ខ្ញុំ + អង្គុយ + លើកៅអី (I sit on the chair)
7	SC	Subordinate Clause	CONJP + S CONJP + VP	នៅពេលខ្ញុំសំអាតបន្ទប់ (when I clean the room.) ប៉ុន្តែអង្គុយចុះ (But sit down)

No	Phrase	Description	Example of Grammar	Example
8	RC	Relative Clause	PREL + VP PREL + S	ដែល + ធ្វើការនៅក្រុមហ៊ុន (who works in the company) ដែល + ខ្ញុំរៀន (which I study)
9	SPKP	Spoken Phrase	INTJ NP + VP + QUES + END	អូ! (Oh!) ខ្ញុំធ្វើបានល្អទេ? (Am I doing a good job?)
10	THEREP	There Verb-to-be Phrase	THEREP + RP THEREBE	មាន + តែ (There is + only) មាន (There is)
11	VP	Verb Phrase	VP + NP VP + NP + RP V + VP V + CP	ផឹកតែ + ទឹកត្រជាក់ (Drink only + cold water) ទម្លាក់ + កាំបិត + ចុះ (Put the knife down) ជួយ + សំអាតបន្ទប់ (help clean the room) ដឹង + ថាគេជាសិស្ស (know that he's a student)
12	CP	Complement Phrase	COMP + S	ជឿជាក់ + ថា + គាត់ធ្វើបាន

Appendix B: Grammar rules with frequency

No	Grammar Rules	Frequency
1	NP -> N	82
2	S -> NP VP	78
3	VP -> VERB NP	66
4	NP -> PRON	46
5	NP -> N PP	20
6	S2 -> NP VP	15
7	PP -> PREP PRON	15
8	S1 -> NP VP	12
9	PP -> PREP NP	11
10	SC -> CONJ S2	9
11	ADJP -> ADJ	8
12	ADVP -> ADV	8
13	VP -> AUX VP	7
14	S -> S1 CONJ S2	7
15	NP -> N ADJ	7
16	PP -> PREP N	7
17	VP -> VERB ADVP	7
18	NP -> N PREP PRON	6
19	NP -> N PDMN	6
20	NP -> N ADJP	6
21	VP -> VERB PP	5
22	ADVP -> FIXADJ ADJP	5
23	VP -> VERB	5
24	VP -> VERB SC	4
25	VP -> VERB N	4
26	NP -> NP PP	4
27	ADJP -> FIXADJ ADJ	4
28	S -> S1	4
29	VP -> VERB NP RP	3
30	NP -> N SC	3
31	ADJP -> ADJ ADVP	3
32	NP -> N ADJ PP	2
33	ADVP -> FIXADV ADV	2

No	Grammar Rules	Frequency
34	NP -> N CDMN ADJP	2
35	VP -> VERB N ADV	2
36	NP -> N N	2
37	NP -> N RC	2
38	NP -> N DONM	2
39	RC -> PREL VP	2
40	VP -> VERB NP SC	2
41	NP -> N PCDM	2
42	NP -> N CONJ N	2
43	NP -> N PRON	2
44	VP -> VERB VP ADVP	2
45	VP -> VERB NP PP	2
46	NP -> N CDMN	2
47	VP -> VERB NP ADVP	2
48	VP -> VERB RP	2
49	NP -> NP CONJ NP	1
50	NP -> NOUN CDMN ADJP	1
51	S -> NP ADJP	1
52	S -> NP	1
53	VP -> V NP	1
54	VP -> VERB VP	1
55	NP -> N DCNM	1
56	VP -> VERB NP ENEG	1
57	NP -> PRON VP	1
58	VP -> VERB CP	1
59	ADJP -> ADVP ADJ	1
60	VP -> ADVP VP	1
61	ADVP -> NEG	1
62	CP -> COMP S1	1
63	VP -> AUX VERB NP	1
64	S2 -> NP ADJP	1
65	NP -> NOUN ADJ	1