PARSING OF MYANMAR SENTENCES WITH FUNCTION TAGGING

Win Win Thant¹, Tin Myat Htwe² and Ni Lar Thein³

¹,³University of Computer Studies, Yangon, Myanmar
winwinthant@gmail.com
nilarthein@gmail.com
²Natural Language Processing Laboratory
University of Computer Studies, Yangon, Myanmar
tinmyathtwe@gmail.com

ABSTRACT

This paper describes the use of Naive Bayes to address the task of assigning function tags and context free grammar (CFG) to parse Myanmar sentences. Part of the challenge of statistical function tagging for Myanmar sentences comes from the fact that Myanmar has free-phrase-order and a complex morphological system. Function tagging is a pre-processing step for parsing. In the task of function tagging, we use the functional annotated corpus and tag Myanmar sentences with correct segmentation, POS (part-of-speech) tagging and chunking information. We propose Myanmar grammar rules and apply context free grammar (CFG) to find out the parse tree of function tagged Myanmar sentences. Experiments show that our analysis achieves a good result with parsing of simple sentences and three types of complex sentences.

KEYWORDS

Function tagging, Parsing, Naive Bayes theory, Context free grammar, Myanmar sentences

1. INTRODUCTION

The natural language processing community is in the strong position of having many available approaches to solve some of its most fundamental problems [1]. We have taken Myanmar language for information processing. Myanmar is an agglutinative language with a very productive inflectional system. This means that for any NLP application on Myanmar to be successful, some amount of functional analysis is necessary. Without it, the development of grammatical relations would not be feasible due to the sparse data problem bound to exist in the training data. Our approach is a part of the Myanmar to English machine translation project. If high quality translation is to be achieved, language understanding is a necessity. One problem in Myanmar language processing is the lack of grammatical regularity in the language. This leads to very complex Myanmar grammar in order to obtain satisfactory results, which in turn increases the complexity in the parsing process, it is desired that simple grammar is to be used.

Our proposed method makes use of two components. They are function tagging and parsing. Function tags are useful for any application trying to follow the thread of the text—they find the ‘who does what’ of each clause, which can be useful to gain information about the situation or to learn more about the behaviour of words in the sentence [2]. The goal of function tagging is to assign syntactic categories like subject, object, time and location to each word in the text document. In case of function tagging, we use Naive Bayes theory and the functional annotated tagged corpus. Parsing is the process of analyzing a text or sentence that is made up of a sequence of words called tokens, and to determine its grammatical structure with respect to a given grammatical rules. The goal of the second one is to produce the parse tree of the sentences in Myanmar text.
In our approach, we take the chunk level phrase with the combination of POS tag and its category which is the output of a fully described morphological analyzer [3][4], which is very important for agglutinative languages like Myanmar. A small corpus annotated manually serves as training data because the large scale Myanmar Corpus is unavailable at present. Since the large-scale annotated corpora, such as Penn Treebank, have been built in English, statistical knowledge extracted from them has been shown to be more and more crucial for natural language disambiguation [5]. As a distinctive language, Myanmar has many characteristics different from English. The use of statistical information efficiently in Myanmar language is still a virgin land waiting to explore.

The rest of the paper is organized as in the followings. Next, in the Related Work section, we analyze previous efforts related to the tasks of function tagging and parsing. Section 3 explains Myanmar language. Section 4 describes about corpus statistics. Section 5 explains the procedure of proposed system. Section 6 includes the function tag sets. Section 7 describes about the proposed grammar for Myanmar language. Function tagging model is presented in section 8. Section 9 describes about parsing of Myanmar simple and complex sentences. Section 10 explains about experimental results. Finally the conclusion of the paper is presented.

2. RELATED WORK

Blaheta and Johnson [6] addressed the task of function tags assignment. They used a statistical algorithm based on a set of features grouped in trees, rather than chains. The advantage was that features can better contribute to overall performance for cases when several features are sparse. When such features are conditioned in a chain model the sparseness of a feature can have a dilution effect of an ulterior (conditioned) one.

Mihai Lintean and Vasile Rus[7] described the use of two machine learning techniques, naive Bayes and decision trees, to address the task of assigning function tags to nodes in a syntactic parse tree. They used a set of features inspired from Blaheta and Johnson [6]. The set of classes they used in their model corresponds to the set of functional tags in Penn Treebank. To generate the training data, they have considered only nodes with functional tags, ignoring nodes unlabeled with such tags. They trained the classifiers on sections 1-21 from Wall Street Journal (WSJ) part of Penn Treebank and used section 23 to evaluate the generated classifiers.

Yong-uk Park and Hyuk-chul Kwon [8] tried to disambiguate for syntactic analysis system by many dependency rules and segmentation. Segmentation is made during parsing. If two adjacent morphemes have no syntactic relations, their syntactic analyzer makes new segment between these two morphemes, and find out all possible partial parse trees of that segmentation and combine them into complete parse trees. Also they used adjacent-rule and adverb subcategorization to disambiguate of syntactic analysis. Their syntactic analyzer system used morphemes for the basic unit of parsing. They made all possible partial parse trees on each segmentation process, and tried to combine them into complete parse trees.

Mark-Jan Nederhof and Giorgio Satta[9] considered the problem of parsing non-recursive context-free grammars, i.e., context-free grammars that generate finite languages and presented two tabular algorithms for these grammars. They presented their parsing algorithm, based on the CYK (Cocke–Younger–Kasami) algorithm and Earley’s algorithm. As parsing CFG (context-free grammar), they have taken a small hand-written grammar of about 100 rules. They have ordered the input grammars by size, according to the number of nonterminals (or the number of nodes in the forest, following the terminology by Langkilde (2000)).

Kyongho Min and William H. Wilson [10] discussed the robustness of four efficient syntactic error-correcting parsing algorithms that are based on chart parsing with a context-free grammar. They implemented four versions of a bottom-up error-correcting chart parser: a basic bottom-up chart parser, and chart parsers employing selectivity, top-down filtering, and a combination of selectivity and a top-down filtering. They detected and corrected syntactic errors using a system
component called IFSCP (Ill-Formed Sentence Chart Parser) described by Min & Wilson (1994), together with a spelling correction module. They tested 4 different lengths of sentences (3, 5, 7, and 11) and 5 different error types, with a grammar of 210 context-free rules designed to parse a simple declarative sentence with no conjunctions, passivisation, or relative clauses.

3. MYANMAR LANGUAGE

Myanmar (formerly known as Burma) is one of the South-East Asian countries. There are 135 ethnic groups living in Myanmar. These ethnic groups speak more than one language and use different scripts to present their respective languages. There are a total of 109 languages spoken by the people living in Myanmar [11]. The Myanmar language is the official language and is more than one thousand years old.

3.1. Features of Myanmar Language

Generally Myanmar sentence follows the subject, object, and verb pattern. However the interchange of subject, object is acceptable. Unlike English language Myanmar is syntax of relatively free-phrase-order language. Myanmar phrases can be written in any order as long as the verb phrase is at the end of sentence. This can be easily illustrated with the example “အောင်လိမ စာအုပ်ကို စားပြဲတွင် ထားသည်” (He places the book on the table) as shown in table 1. All are valid sentences [12].

| Case   | Myanmar Sentences | Word order               |
|--------|-------------------|--------------------------|
| Case 1 | သူ စာအုပ်ကို စားပြဲတွင် ထားသည် | (Subj-Obj-Pla-Verb)     |
| Case 2 | စာအုပ်ကို စားပြဲတွင် ထားသည် | (Subj-Pla-Obj-Verb)     |
| Case 3 | စာအုပ်ကို စားပြဲတွင် သူ ထားသည် | (Obj-Pla-Subj-Verb)     |
| Case 4 | စာအုပ်ကို သူ စားပြဲတွင် ထားသည် | (Obj-Subj-Pla-Verb)     |
| Case 5 | စာအုပ်ကို သူ စားပြဲတွင် ထားသည် | (Pla-Subj-Obj-Verb)     |
| Case 6 | စားပြဲတွင် သူ စာအုပ်ကို ထားသည် | (Pla-Obj-Subj-Verb)     |

In all the cases, subject is သူ (He), object is စာအုပ်ကို (the book), place is စားပြဲတွင် (on the table) and verb is ထားသည် (places). From the above example, it is clear that phrase order does not determine the functional structure in Myanmar language and permits scrambling. Myanmar language follows Subject-Object-Verb orders in contradiction with English language.

3.2. Issues of Myanmar Language

The highly agglutinative language like Myanmar, nouns and verbs get inflected. Many times we need to depend on syntactic function or context to decide upon whether the particular word is a noun or adjective or adverb or post position [12]. This leads to the complexity in Myanmar grammatical relations. A noun may be categorized as common, proper or compound. Similarly, verb may be finite, infinite, gerund or contingent.

A number of issues are affecting the function tagging for Myanmar language.

- The subject or object of the sentence can be skipped, and still be a valid sentence.
  For example:
  ရန္ကုန္ - သို႔ - သြားသည္ (Go to Yangon)
• Myanmar language makes prominent usage of particles, which are untranslatable words that are suffixed or prefixed to words to indicate level of respect, grammatical tense, or mood. For example:

Myanmar particle - first - prize - wins - if - his parents - PPM - surprise - will
(If Mr. Mg wins the first prize, his parents will surprise.)

• In Myanmar language, an adjective can specialize before or after a noun unlike other languages. For example:

- (or)

(He is a rich man.)

• The subject/object can be another sentence, which does not contain subject or object. For example:

I see the children playing under the tree.

• The postpositions of time phrases or place phrases can be omitted. For example:

She goes to school.

These issues will cause a lot of problem during function tagging, and a lot of possible tags will be resulted.

3.3. Grammar of Myanmar Language

Grammar studies the rules behind languages. The aspect of grammar that does not concern meaning directly is called syntax. Myanmar (syntax: SOV), because of its use of postposition (wi.Bat), would probably be defined as a “postpositional language”, whereas English (syntax: SVO) because of its use of preposition would probably be defined as a “prepositional language”.

There are really only two parts of speech in Myanmar, the noun and the verb, instead of the usually accepted eight parts (Pe Maung Tin 1956:195). Most Myanmar linguists [13] accepted there are eight parts of speech in Myanmar. Myanmar nouns and verbs need the help of suffixes or particles to show grammatical relations.
Myanmar is a highly verb-prominent language and that suppression of the subject and omission of personal pronouns in connected text result in a reduced role of nominals. This observation misses the critical role of postposition particles marking sentential arguments and also of the verb itself being so marked. The key to the view of Myanmar being structures by nominals is found in the role of the particles. Some particles modify the word's part of speech. Among the most prominent of these is the particle အ, which is prefixed to verbs and adjectives to form nouns or adverbs. There is a wide variety of particles in Myanmar [14].

For example:
သူတို႔သည္ မႏ ၱေလးတြင္ ၈ ရက္ တိတိ လည္ခဲ့သည္။

Stewart remarked that "The Grammar of Burmese is almost entirely a matter of the correct use of particles" (Stewart 1956: xi). How one understands the role of the particles is probably a matter of one's purpose.

**3.4. Syntactic Structure of Myanmar Language**

It is known that many postpositions can be used in a Myanmar sentence. If the words can be misplaced in a sentence, the sentence can be abnormal. There are two kinds of sentence as a sentence construction. They are simple sentence (SS) and complex sentence (CS). In simple sentence, other phrases such as object, time, and place can be added between subject and verb. There are two kinds of clause in a complex sentence called independent clause (IC) and dependent clause (DC). There must be at least one independent clause in a sentence. But there can be more than one dependent clause in it. IC contains sentence’s final particle (sfp) at the end of a sentence [15].

\[
\begin{align*}
\text{SS} &= \text{IC}+\text{sfp} \\
\text{CS} &= \text{DC}...+\text{IC}+\text{sfp}
\end{align*}
\]

IC may be noun phrase or verb or combination of both.

\[
\begin{align*}
\text{IC} &= \text{N}... \\
\text{IC} &= \text{V} \\
\text{IC} &= \text{N}...+\text{V}
\end{align*}
\]

DC is the same as IC but it must contain a clause marker (cm) in the end.

\[
\begin{align*}
\text{DC} &= \text{N}...+\text{cm} \\
\text{DC} &= \text{V}+\text{cm} \\
\text{DC} &= \text{N}...+\text{V}+\text{cm}
\end{align*}
\]

**4. CORPUS STATISTICS**

Corpus is a large and structured set of texts. It is used to do statistical analysis, checking occurrences or validating linguistic rules on a specific universe. Besides, it is a fundamental basis of many researches in Natural Language Processing (NLP). Building of the corpus will be helpful for development NLP tools (such as grammar rules, spelling checking, etc). However, there are very few creations and researches of corpora in Myanmar, comparing to other language such as English.

We collected several types of Myanmar texts to construct a corpus. Our corpus is to be built manually. We extended the POS tagged corpus that is proposed in [3]. The chunk and function
tags are manually added to the POS tagged corpus. The number of sentences is about 3900 sentences with average word length 15 and it is not a balanced corpus that is a bit biased on Myanmar textbooks of middle school. The corpus size is bigger and bigger because the tested sentences are automatically added to the corpus. In table 2, Myanmar grammar books and websites are text collections. Example corpus sentence is shown in figure 1.

Table 2. Corpus statistics

| Text types                        | # of sentences |
|-----------------------------------|----------------|
| Myanmar textbooks of middle school| 1250           |
| Myanmar Grammar books             | 628            |
| Myanmar Newspapers                | 730            |
| Myanmar websites                  | 970            |
| Others                            | 325            |
| Total                             | 3903           |

Figure 1. A sentence in the corpus

5. PROPOSED SYSTEM

The procedure of the proposed approach is shown in the following figure.

Accept input Myanmar sentence with segmentation, POS tagging and chunking
↓
Extract one POS tag and its category from each chunk
↓
Choose the possible function tags for each POS tag by using Naive Bayes Theory
↓
Display the sentence with function tags
↓
Parse the function tags by using CFG rules with the proposed grammar
↓
Display the parse tree as an output

Figure 2. Proposed system

6. FUNCTION TAGSET

Function tagging is a process of assigning syntactic categories like subject, object, time and location to each word in the text document. These are conceptually appealing by encoding an event in the format of “who did what to whom, where, when”, which provides useful semantic
information of the sentences. We use the function tags that is proposed in [16] because it is easier to maintain and can add new language features. The function tagset is shown in table 3.

Table 3. Function tagset

| Tag     | Description          | Example                  |
|---------|----------------------|--------------------------|
| Active  | Verb                 | စားသည္                   |
| Subj    | Subject              | သူ                        |
| PSubj   | Subject              | သူ                        |
| SubjP   | Postposition of Subject | သည္                    |
| Obj     | Object               | ေကာ္ဖီ                     |
| PObj    | Object               | ေကာ္ဖီ                     |
| ObjP    | Postposition of Object | ကို                   |
| PlObj   | Indirect Object      | အား                         |
| IobjP   | Postposition of Indirect Object | စီး          |
| Pla     | Place                | အနက္                                     |
| PPla    | Place                | မင္းသမီး                   |
| PlaP    | Postposition of Place | သို႔မနက္                     |
| Tim     | Time                 | မန္း                              |
| PTim    | Time                 | စီး                              |
| TimP    | Postposition of Time | မနက္                      |
| PExt    | Extract              | ထြန္းထြန္း                         |
| ExtP    | Postposition of Extract | ဟု                    |
| PSim    | Similie              | မိုး                              |
| SimP    | Postposition of Similie | သူသည္ဆရာ                     |
| PCom    | Compare              | သူသည္ဆရာ                     |
| ComP    | Postposition of Compare | ဟု                        |
| POwn    | Own                  | အန္                          |
| OwnP    | Postposition of Own  | အတြက္                      |
| Ada     | Adjective            | လွအတြက္                                  |
| PcompS  | Subject Complement   | နည္းကိုစစ္စူး                |
| PcompO  | Object Complement    | လက္စြပ္ သည္                         |
| PcompIO | Object Complement    | ထြန္းထြန္း                         |
| PcompIOP| Postposition of Object Complement | ဟု            |
| Puse    | Use                  | ခြင့္                                    |
| UseP    | Postposition of Use  | စီး                          |
| PCau    | Cause                | တုတ္                                             |
| CauP    | Postposition of Cause | မိုး                        |
| PAim    | Aim                  | အတြက္                        |
| AimP    | Postposition of Aim  | အတြက္                        |
| CCS     | Join the sentences   | ကို                          |
| CCM     | Join the meanings    | ထြန္းထြန္း                         |
| CCC     | Join the words       | မိုး                          |
| CCP     | Join with particles  | သူသည္ဆရာ                     |
| CCA     | Join as an adjective | ဟု                          |
7. PROPOSED GRAMMAR FOR MYANMAR SENTENCES

Since it is impossible to cover all types of sentences in Myanmar language, we have taken some portion of the sentence and try to make grammar for them. Myanmar is free-phrase-order language. In Myanmar language, we see that one sentence can be written in different forms for the same meaning, i.e. the positions of the tags are not fixed. So we cannot restrict the grammar rule for one sentence. The grammar rule may be very long, but we have to accept it. The grammar rule we have tried to make, may not work for all the sentences in Myanmar language because we have not considered all types of sentences. Some of the sentences are shown below, which are used to make the grammar rules.

- သူ-သည္-ေက်ာင္း-သို႔-သြား-သည္။ (Subj-Pla-Verb)
- သူ-သည္-ေက်ာင္းသား-တစ္ေယာက္-ျဖစ္-သည္။ (Subj-PcomplS-Verb)
- ေကာင္စီ-၀င္-အျဖစ္-သူ-ကို-လူထု-က-ေရြး-သည္။ (Subj-Obj-Use-Verb)
- ေမာင္လွ-သည္-ေခြး-ကို-တုတ္-ျဖင့္-ရိုက္-သည္။ (Subj-Obj-Use-Verb)
- သူ-သည္-ဆရာ႔-ကို-စာအုပ္-ေပး-သည္။ (Subj-Obj-Iobj-Verb)
- သူမ-သည္-လူနာမ်ား-ကို-ေဆြမ်ိဳးမ်ား-ကဲ႔-သို႔-ျပဳစု-သည္။ (Subj-Obj-Sim-Verb)
- ကေလးမ်ား-သည္-အေဖာ္-ေၾကာင့္-ပ်က္စီး-သည္။ (Subj-Cau-Verb)
- သစ္႐ြက္-တို႔-သည္-တေပါင္း-လ-၌-ေၾကြ-သည္။ (Subj-Tim-Verb)
- တရားသူ-ၾကီး-သည္-ခိုးမႈ-ကို-တရား႐ံုး-၌-နံနက္-က-စစ္ေဆး-သည္။ (Subj-Obj-Pla-Tim-Verb)
- အေမ-သည္-သူ-သား-အတြက္-မုန္႔-ကို-ေစ်း-မွ-မနက္-က-ဝယ္ခဲ႔-သည္။ (Subj-Aim-Obj-Pla-Tim-Verb)

Our proposed grammar for Myanmar Sentences:

Sentence → I-sent | I-sent CC I-sent | CCM I-sent | Obj-sent I-sent | Subj-sent I-sent
I-sent → Subj Obj Pla Active | Subj Active | Com Pla Active | Subj PcomplIS Active
CC → CCS | CCP
Subj -sent → I-sent CCA Subj
Obj -sent → I-sent CCA Obj
Subj → PSubj SubjP
Subj → Subj
Obj → PObj ObjP
Obj → Obj
Pla → PPla PlaP
PcomplO → PPcomplO PcomplOP
Use → PUse UseP
Sim → PSim SimP

8. FUNCTION TAGGING

8.1 Naive Bayes Classifier

Before one can build naive Bayesian based classifier, one needs to collect training data. The training data is a set of problem instances. Each instance consists of values for each of the defined features of the underlying model and the corresponding class, i.e. function tag in our case. The development of a Naïve Bayes classifier involves learning how much each function tag should be trusted for the decisions it makes [17]. It is well-matched to the function tagging problem.

The Naïve Bayesian classifier is a term in Bayesian statistics dealing with a simple probabilistic classifier based on applying Bayes’ theorem with strong (naïve) independence assumptions. It assumes independence among input features. Therefore, given an input vector, its target class can be found by choosing the one with the highest posterior probability. The probability model for a classifier is a conditional model.
\[ P(c_k| x_1, x_2, \ldots, x_i) = P(c_k) \times P(x_1, x_2, \ldots, x_i | c_k) \]  
(1)

Let \( X=x_1, x_2, x_3, \ldots (x_i, i \geq 1 \text{ and } X \text{ are features}) \)
\( C=c_1, c_2, c_3, \ldots (c_k, k \geq 1 \text{ and } C \text{ are classes}) \)
\( P(c_k| x_1, x_2, \ldots, x_i) \) is referred to as the posterior probability
\( P(c_k) \) as the prior probability
\( P(x_1, x_2, \ldots, x_i | c_k) \) as the log likelihood

8.2. Function Tagging by Using Naïve Bayes Theory

The labels such as subject, object, time, etc. are named as function tags. By function, it is meant that action or state which a sentence describes. The system operates at word-level with the assumption that input sentences are pre-segmented, pos-tagged and chunked.

Each proposed function tag is regarded as a class and the task is to find what class/tag a given word in a sentence belongs to a set of predefined classes/tags. A feature is a POS tag word with category. The category of a word is added to the POS tag to obtain more accurate lexical information. It can be formed from the features of that word.

For example: Ma Ma is a clever student.

\[ \text{Ma Ma} [\text{မမ(n.person)} \text{ သည္(ppm.subj)}] \text{ clever [တူဆိုင္(adj.dem)]} \text{ student [သည္(n.person)]} \]
\[ \text{a [တစ္(part.number) သည္(ppm.type)]} \text{ is [သည္(v.common) သည္(sf.declarative)]} \]

Noun has 16 categories such as animals, person, objects, food, location, etc. There are 47 categories in our corpus. We show some features of Myanmar words as shown in table 4.

| Feature     | English | Myanmar |
|-------------|---------|---------|
| n.food      | apple   | ပန္းသီး |
| pron.posessive | his     | သူ |
| ppm.time    | at      | တြင္ |
| adj.dem     | happy   | ေပ်ာ္ရႊင္ေသာ |
| part.support| can     | ႏိုင္ |
| cc.mean     | so      | ထို႔ေၾကာင့္ |
| v.common    | go      | သြား |
| sf.declarative | null   | ၏ |

In Myanmar language, some words have same meaning but in different features as shown in table 5. For example:

- Ma Ma and Hla Hla are friends.
- He lives with his uncle.
- He hits the dog with the stick.

In these three sentences, English words (and, with, with) have the same Myanmar meaning (ႏွင့္).

| Feature    | English | Myanmar |
|------------|---------|---------|
| cc.chunk   | and     | ႏွင့္ |
| ppm.compare | with   | ႏွ |
| ppm.use    | with    | ႏွ |
A class is one of the proposed function tags. Same word may have different function tags as shown in Table 6.

Table 6. Function tags

| Function tags | English | Myanmar |
|---------------|---------|---------|
| PcomplS       | He has a **house**. | အိမ္ |
| PPla          | He lives in a **house**. | အိမ္ |
| PSubj         | A **house** is near the school. | အိမ္ |
| PObj          | He buys a **house**. | အိမ္ |

There are many chunks in a sentence such as NC (noun chunk), PPC (postpositional chunk), AC (adjectival chunk), RC (adverbial chunk), CC (conjunctural chunk), SFC (sentence’s final chunk) and VC (verb chunk). The chunk types are shown in Table 7.

Table 7. Chunk types

| No. | Chunk Type         | English | Example                              |
|-----|--------------------|---------|--------------------------------------|
| 1   | Noun Chunk         | they    | NC[ဗိ/pron.person]                  |
| 2   | Postpositional Chunk | at     | PPC[တေ/ppm.place]                  |
| 3   | Adjectival Chunk   | brave   | AC[rရဲ/adj.dem]                    |
| 4   | Adverbial Chunk    | quickly | RC[လ်င္ျမန္/adv.manner]            |
| 5   | Conjunctural Chunk | or      | CC[သို႔မ/cc.chunk]                 |
| 6   | Sentence Final Chunk | -      | SFC[၏/sf.declarative]              |
| 7   | Verb Chunk         | help    | VC[ကူညီ/v.common]                 |

A chunk contains a Myanmar head word and its modifier. It can contain more than one POS tag and one of the POS tags is selected with respect to the chunk type. In the following chunk, the POS tag (n.animals) is selected with respect to the chunk type (NC).

For example: NC[ဗိ/n.animals, က/adj, တစ္/part.number, ဗိ/part.type]

If the noun chunk (NC) contains more than one noun, the last noun (n.food) is selected as a main word according to the nature of Myanmar language.

For example: NC[င္ျပြီး/n.time, မ်ား/part.number]

There are many possible function tags ($t_1, t_2\ldots t_k$) for each POS tag with category (pc). These possible tags are retrieved from the training corpus by using the following equation that is prior probability as shown in Table 8.

\[ P (t_k|pc) = C (t_k,pc)/C(pc) \]

(2)

Table 8. Sample data for POS/function tag pairs with probability

| POS tags    | Function tags : Probability               |
|-------------|------------------------------------------|
| ppm.use     | UseP:1.0                                  |
| n.natural   | PSubj:0.209, Subj:0.2985, PPla:0.1343, PObj:0.1642, PcomplS:0.0448, PPcomplO:0.0149, PCau:0.0448, PSim:0.0149, PAim:0.0299, |
We calculate the probability between next function tags \( (n_j, n_k) \) and previous possible tags by using the following equation that is log likelihood as shown in Table 9.

\[
P(n_j|t_k) = \frac{C(n_j, t_k)}{C(t_k)}
\]  \( (3) \)

Table 9. Sample data for function/function tag pairs with probability

| Function tags | Function tags : Probability |
|--------------|-----------------------------|
| CCC          | Subj:0.271, Active:0.2452 , POBJ:0.1226, Obj:0.129, Ptim:0.0194 PcomplS:0.0516, PPla:0.0516, Pla:0.0387, Tim:0.0194, PSubj:0.0387 PCau:0.0065, Paim:0.0065 |
| Subj         | CCC:0.2047, Active:0.5436, Ptim:0.0067, PCom:0.0067, Ada:0.0604, PDir:0.0067, Tim:0.0134, Pla:0.0101, PUse:0.0034, Sim:0.0101, PLea:0.0134, CCA:0.0268, Obj:0.0503, PPla:0.0235, Pobj:0.0168 CCS:0.0034 |
| PCau         | CCC:0.1111, CauP:0.8889 |
| Pext         | ExtP:1.0 |
| UseP         | Active:0.5652, POBJ:0.087, subj:0.087, PArr:0.0435, Ptim:0.087, CCA:0.0435, PcomplS:0.0435, Obj:0.0435 |
| PPla         | CCC:0.056, PlaP:0.936, PPla:0.0080 |
| Obj          | CCC:0.2667, Active:0.6917, AimP:0.0083, subj:0.0083, CCA:0.0083 Ada:0.0167 |
| PcomplIO     | Active:1.0 |

Possible function tags are disambiguated by using Naïve Bayesian method. We multiply the probabilities from (2) and (3) and choose the function tag with the largest number as the posterior probability.

Technically, the task of function tags assignment is to generate a sentence that has correct function tags attached to certain words. Our description of the function tagging process refers to the example as shown in figure 3, which illustrates the sentence (“မမမဘုံ၊လုသည္ ဗက္းသို႔ စက္႔ဖင့္ သြားသည္” (Ma Ma and Hla Hla go to school by bicycle). This sentence is represented as a sequence of word-tags as “noun verb conjunction noun ppm pronoun verb”. It is described as a sequence of chunk as “NC VC CC NC PPC NC VC SFC”.

Figure 3. An overview of function tagging of the sentence
(a) The input POS-tagged and chunk sentence (b) The output sentence with function tags
9. Parsing

9.1. Context Free Grammar for Myanmar Sentences

The LANGUAGE defined by a CFG (context-free grammar) is the set of strings derivable from the start symbol S (for Sentence). The core of a CFG grammar is a set of production rules that replaces single variables with strings of variables and symbols. The grammar generates all strings that, starting with a special start variable, can be obtained by applying the production rules until no variables remain. A CFG is usually thought in two ways: a device for generating sentences, or a device if assigning a structure to a given sentence. We use CFG for grammatical relations of function tags.

A CFG is a 4-tuple \(<N, \Sigma, P, S>\) consisting of

- A set of non-terminal symbols \(N\)
- A set of terminal symbols \(\Sigma\)
- A set of productions \(P\)
- A designated start symbol \(S\)

9.2. Parsing Simple Sentences

A simple sentence contains one subject and one verb. We can construct simple sentences in many different forms.

- Constructed by adding adjective and adverb
  
  Adjective + Subject + Adjective + Object + Adverb + Verb
  
  ဝေသာ +ေကာင္ေလးသည္ +ခ်ိဳေသာ +ကိတ္မုန္႔ကို +လ်င္ျမန္စြာ +စားသည္။

  (A fat boy eats quickly the sweet cake.)

- Constructed by using different set of phrases
  
  Subject phrase + Object phrase + Verb
  
  ဦးဘ၏သားသည္ +ဦးထုပ္အနီႏွင့္ေကာင္ေလးကို +ရွာသည္။

  (U Ba’s son finds a boy with the red hat.)

- Constructed by omitting subject
  
  Object + Time + Verb
  
  ဆံပင္ကို +တနဂၤေႏြေန႔တြင္ +ေလွ်ာ္သည္။

  (Wash the hair in Sunday.)

- Constructed by omitting verb
  
  Subject + Subject’s complement + Sentence’s final particle
  
  သူက +ဆရာ +null

  (He is a teacher.)

Consider a simple declarative sentence “သူတို႔သည္ ေမာင္ဘကို ေခါင္းေဆာင္ အျဖစ္ ေရြးခ်ယ္ခဲ့ သည္။” (They selected Mg Ba as a leader).

The structure of the above sentence is Subj-Obj-PcomplO-Active. This is a correct sentence according to the Myanmar literature.
9.3. Parsing Complex Sentences

Complex sentence has more than one verb. It contains at least two simple sentences. Simple sentences are joined with postpositions, particles or conjunctions. There are three types of complex sentences.

9.3.1. Two simple sentences are joined with postpositions

Consider a complex sentence "ဗိုလ်ချွေးကြပ်သည် ကို ကျွန်းသည်" (I see that he is swimming).

In this sentence, two simple sentence ဗိုလ်ချွေးကြပ်သည် (he is swimming) and ကျွန်းသည် (I see) is joined by postposition ကို (that). The structure of the above sentence is Subj-Active-CCP-Subj-Active. This is a correct sentence according to the Myanmar literature.

(a) \( NC(\text{ဗိုလ်ချွေး/pron.person}) \rightarrow VC(\text{ကြပ်သည်/v.common}) \rightarrow CC(\text{ကို/cc.obj}) \rightarrow NC(\text{ကျွန်းသည်/pron.person}) \rightarrow VC(\text{သည်/v.common}) \rightarrow SFC(\text{သည်/sf}) \)

(b) \( \text{Subj}(\text{ဗိုလ်ချွေး}) \rightarrow \text{Active}(\text{ကြပ်သည်}) \rightarrow \text{CCP}(\text{ကို}) \rightarrow \text{Subj}(\text{ကျွန်းသည်}) \rightarrow \text{Active}(\text{သည်}) \)

(c) \( \text{Sentence} \rightarrow \text{I-sent CCP I-sent} \)

\( \text{I-sent CCP I-sent} \rightarrow \text{Sentence} \rightarrow \text{Subj Active CCP I-sent} \)

\( \text{Subj Active CCP I-sent} \rightarrow \text{I-sent} \rightarrow \text{Subj Active} \)

\( \text{Subj Active CCP Subj Active} \rightarrow \text{I-sent} \rightarrow \text{Subj Active} \)
9.3.2. Two simple sentences are joined with particles

In figure 7, the sentence “အေဖေပးေသာစာအုပ္သည္ ေကာင္းသည္” (The book that is given by my father is good.) is illustrated. It is described as a sequence of chunk as “NC VC CC NC PPC AC SFC” and the sentence structure (Sentence) contains separate constituents for the subject sentence (Subj-sent) and independent sentence (I-sent), which contains other phrases.

(a) NC [နောင်/n.person] # VC [ထောင်/v.common] # CC [သည်/cc.adj] # NC [စာအုပ္/n.objects] # PPC [ဆိုတင်/ppm.subj] # AC [ဗျဲ/adj.dem] # SFC [သည်/sf]

(b) Subj[နောင်]#Active[ထောင်]#CCA[သည်]#PObj[စာအုပ္]#ObjP[သည်]#Active[ဗျဲ]

(c)

Sentence
Subj-sent I-sent [Sentence→Subj-sent I-sent]
I-Sent CCA Subj I-sent [Subj-sent→I-Sent CCA Subj]
Subj Active CCA Subj I-sent [I-sent→Subj Active]
Subj Active CCA PSubj SubjP I-sent [Subj→PSubj SubjP]
Subj Active CCA PSubj SubjP Ada [I-sent→Ada ]
9.3.3. Two simple sentences are joined with conjunctions

Consider a complex sentence “ကြီးမှာ တောင်သား သူငယ် ကြယ်စုတန်း” (As he is clever, the teachers love him). In this sentence, two simple sentence ကြီးမှာ (he is clever) and သူငယ် ကြယ်စုတန်း (the teachers love him) is joined by postposition တောင်သား (as). The structure of the above sentence is Subj-Ada-CCS- Subj-Obj-Active. This is a correct sentence according to the Myanmar literature.

(a) NC [။/pron.person] # AC [။။/adj.dem] # CC [။။။/cc.sent] # NC [။။/n.objects] # PPC [။/ppm.subj] # NC [။/pron. possessive] # PPC [။/ppm.obj] # VC [။/v.common] # SFC [။/sf]

(b) Subj[။]#Ada[။]#CCS[။။/cc.sent]#PSubj[။]#SubjP[။]#PObj [။/pron. possessive] # ObjP [။/ppm.obj] # VC [။/v.common] # SFC [။/sf]

(c) Sentence [start]
   I-sent CCS I-sent
   Subj Ada CCS I-sent
   Subj Ada CCS Subj Obj Active
   Subj Ada CCS PSubj SubjP Obj Active
   Subj Ada CCS PSubj SubjP PObj ObjP Active
   [Obj → PObj ObjP]

(d)

Figure 7. (a) The tagged and chunk complex sentence joined with conjunction (CCS)
   (b) The function tagged sentence (c) Grammar derivation (d) The parse tree with function tags

10. EXPERIMENTAL RESULTS

In our corpus, all sentences can be further classified as two sets. One is simple sentence set, in which every sentence has no more than 15 words. The other is complex sentence set, in which every sentence has more than 15 words. There are 1600 simple sentences and 2300 complex sentences in the corpus.

For evaluation purpose, different numbers of sentences collecting from Myanmar textbooks of middle school and Myanmar historical books are used as a test set. There are about 2200 sentences in the test set. After implementation of the system using the grammar, it has been seen that the system can easily generates the parse tree for a sentence if the sentence structure satisfies the grammar rules. Our program tests only the sentence structure according to the
grammar rules. So if the sentence structure satisfies the grammar rule, program recognizes the sentence as a correct sentence and generates a parse tree. Otherwise it gives output as an error.

Table 10 shows the overall performance for the proposed system. The proposed system yield 96.68% of precision, 93.05% of recall and 94.83% of f-measure for simple sentence. Performance comparisons between the various sentence types are shown in figure 8.

\[
\text{Precision} = \frac{\text{NumberOfCorrectSentences}}{\text{TotalNumberOfCorrectSentences}} \times 100\%
\]

\[
\text{Recall} = \frac{\text{NumberOfCorrectSentences}}{\text{NumberOfActualExistingCorrectSentences}} \times 100\%
\]

\[
\text{F-Measure} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

Table 10. Compared results of each sentence types

| Sentence Type            | Actual | Recognized | Correct | Precision | Recall | F-Measure |
|--------------------------|--------|------------|---------|-----------|--------|-----------|
| Simple                   | 720    | 693        | 670     | 96.68%    | 93.05% | 94.83%    |
| Complex joined with CCP  | 455    | 420        | 394     | 93.81%    | 88.54% | 91.09%    |
| Complex joined with CCA  | 370    | 351        | 319     | 90.88%    | 86.22% | 88.48%    |
| Complex joined with CCS  | 665    | 640        | 593     | 92.66%    | 89.17% | 90.88%    |

Figure 8. Performance Comparisons between the Various Sentence Types

11. CONCLUSION AND FUTURE WORK

In the task of assigning function tag, we chose Naïve Bayes model for its simplicity and user-friendliness. We apply context-free grammar for parsing because it is easier to maintain and can add new language features. The parse tree can be built by using
function tags. As function tagging is a pre-processing step for parsing, the errors occurred in the task of function tagging affect the parse tree. The corpus may be balanced because Naïve Bayesian framework probability simply describes uncertainty. The corpus creation is time consuming. The corpus is the resource for the development of Myanmar to English translation system and we expect the corpus to be continually expanded in the future because the tested sentence can be added into the corpus.

In this work we have considered limited number of Myanmar sentences to construct the grammar rules. In future work we have to consider as many sentences as we can and some more tags for constructing the grammar rules because Myanmar language is a free-phrase-order language. Word position for one sentence may not be same in the other sentences. So we can not restrict the grammar rules for some limited number of sentences.

REFERENCES

[1] John C. Henderson and Eric Brill. “Exploiting Diversity in Natural Language Processing: Combining Parsers”.
[2] Blaheta, D (2003) “Function tagging”. Ph.D. Dissertation, Brown University. Advisor-Eugene Charniak.
[3] Phyu Hnin Myint (2010) “Assigning automatically Part-of-Speech tags to build tagged corpus for Myanmar language”, The Fifth Conference on Parallel Soft Computing, Yangon, Myanmar.
[4] Phyu Hnin Myint (2011) “Chunk Tagged Corpus Creation for Myanmar Language”. In Proceedings of the ninth International Conference on Computer Applications, Yangon, Myanmar.
[5] Eugene Charniak (1997) “Statistical parsing with a context-free grammar and word statistics”. In Proceedings of the Fourteenth National Conference on Artificial Intelligence, pages 598-603, Menlo Park.
[6] Blaheta, D., and Johnson, M (2000) “Assigning function tags to parsed text”. In Proceedings of the 1st Annual Meeting of the North American Chapter of the Association for Computational Linguistics, 234–240.
[7] Mihai Lintean and Vasile Rus (2007) “Naive Bayes and Decision Trees for Function Tagging”. In Proceedings of the International Conference of the Florida Artificial Intelligence Research Society (FLAIRS) 2007, Key West, FL, May (in press).
[8] Yong-uk Park and Hyuk-chul Kwon (2008) “Korean Syntactic Analysis using Dependency Rules and Segmentation “, Proceedings of the Seventh International Conference on Advanced Language Processing and Web Information Technology(ALPIT2008), Vol.7, pp.59-63, China, July 23-25.
[9] Mark-Jan Nederhof and Giorgio Satta (2002) “Parsing Non-Recursive Context-Free Grammars”. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL ANNUAL’02), July 7-12, Pages 112-119, Philadelphia, Pennsylvania, USA.
[10] Kyongho Min & William H. Wilson, “Are Efficient Natural Language Parsers Robust?” School of Computer Science & Engineering,University of New South Wales, Sydney NSW 2052 Australia
[11] Ethnologue (2005) Languages of the World, 15th Edition, Dallas, Tex.: SIL International. Online version: http://www.ethnologue.com/. Edited by Raymond G. Gordon, Jr.
[12] Myanmar Thudda (1986) vol. 1 to 5 in Bur-Myan, Text-book Committee, Basic Edu., Min. of Edu., Myanmar, ca.
[13] Shwe Pyi Soe, U (2010) Aspects of Myanmar Language
[14] Thaung Lwin, U (1978) ျမန္မာဘာသာစကား အဆိုးပြောင်မှုများ ကမ်းစံမှာခွဲရောက် ျမန္မာဘာသာစကား Aspects of Myanmar Language
Ko Lay, U (2003) Ph.D. Dissertation, Myanmar Department, University of Education.

Win Win Thant (2010) “Naive Bayes for function tagging in Myanmar Language”, The Fifth Conference on Parallel Soft Computing, Yangon, Myanmar, 2010.

Leon Versteegen (1999) “The Simple Bayesian Classifier as a Classification Algorithm”.

Y. Tsuruoka and K. Tsujii (2005) “Chunk parsing revisited”. In Proceedings of the Ninth International Workshop on Parsing Technologies. Vancouver, Canada.

Michael Collins (1996) “A New Statistical Parser Based on Bigram Lexical Dependencies”. In Proceedings of ACL-96, pp. 184–191.

Authors

Win Win Thant is a Ph.D research student. She received B.C.Sc (Bachelor of Computer Science) degree in 2003, B.C.Sc (Hons.) degree in 2004 and M.C.Sc (Master of Computer Science) degree in 2007. She is also an Assistant Lecturer of U.C.S.Y (University of Computer Studies, Yangon). She has published papers in International conferences and International Journals. Her research interests include Natural Language Processing, Artificial Intelligence and Machine Translation.

Tin Myat Htwe is an Associate Professor of U.C.S.Y. She obtained Ph.D degree of Information Technology from University of Computer Studies, Yangon. Her research interests include Natural Language Processing, Data Mining and Artificial Intelligence. She has published papers in International conferences and International Journals.

Ni Lar Thein is a Rector of U.C.S.Y. She obtained B.Sc. (Chem.), B.Sc. (Hons) and M.Sc. (Computer Science) from Yangon University and Ph.D. (Computer Engg.) from Nanyang Technological University, Singapore in 2003. Her research interests include Software Engineering, Artificial Intelligence and Natural Language Processing. She has published papers in International conferences and International Journals.