Function Tagging for Myanmar Language

Win Win Thant
University of Computer Studies, Yangon, Myanmar

Tin Myat Htwe
University of Computer Studies, Yangon, Myanmar

Ni Lar Thein
University of Computer Studies, Yangon, Myanmar

ABSTRACT
Function tagging is one of the essential steps in Myanmar to English machine translation system. In this paper we propose a set of function tags for Myanmar and address the question of assigning function tags to Myanmar words. A small functional annotated tagged corpus manually serves as the training data because the large scale Myanmar Corpus is unavailable at present. Part of the challenge of statistical function tagging for Myanmar language comes from the fact that Myanmar has free phrase order and a complex morphological system. In the task of function tagging, we use the output of morphological analyzer which tags the function of Myanmar sentences with correct segmentation, POS (part-of-speech) tagging [1] and chunking information [2]. We use Naïve Bayesian statistics to disambiguate the possible function tags of each word in the sentence. Function tagging can be exploited by NLP applications such as syntactic and semantic analysis, information retrieval and machine translation. Experiments show that our analysis achieves a good result with simple sentences and complex sentences.

General Terms
Natural Language Processing, Machine Translation

Keywords
Function tagging, function tagset, Myanmar language, functional annotated tagged corpus

1. INTRODUCTION
The natural language processing community is in the strong position of having many available approaches to solving some of its most fundamental problems. The corpus-based statistical function tagging is very essential for Myanmar to English machine translation system. A word can appear in a sentence for two reasons: because it serves a syntactic function, or because it provides semantic content. Words that play different roles are treated differently in human language processing: function and content words produce different patterns of brain activity, and have different developmental trends. To the authors' knowledge, this is the first attempt to do the task of function tagging on Myanmar.

Myanmar is an agglutinative language with a very productive inflectional system. It is also a variable phrase order language. This means that for any NLP applications on Myanmar to be successful, some amount of functional analysis is necessary. Function tagging 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 term increases the complexity in the function tagging, it is desired that simple grammar is to be used. However, this will cause ambiguities in the result.

The system operates at word-level with the assumption that input sentences are pre-segmented, pos-tagged and chunked. Function tags of Myanmar language is defined because these 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 behavior of words in the sentence [3].

A small corpus annotated manually serves as the 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 [4]. 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.

Naïve Bayesian is chosen for its simplicity and user-friendliness. Naive-Bayesian classifier makes strong assumptions about how the data is generated, and use a probabilistic model that reflects the assumptions [5]. They use a collection of labeled training examples to estimate the parameters of the generative model. Classification of new examples is performed with Bayes' rule by selecting the class that is most likely to have generated the example.

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. Section 3 explains about Myanmar language. Section 4 deals with the proposed function tagset. Section 5 describes about corpus creation. Function tagging model is presented in section 6. Section 7 discusses about evaluation method. Section 8 describes about error analysis. Finally the conclusion of the paper is presented.

2. RELATED WORK
Previous work to address the task of function tags assignment is presented in [6]. They use a statistical algorithm based on a set of features grouped in trees, rather than chains. The advantage is 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.

Don Blaheta [7] presented a system that utilizes a maximum-entropy inspired algorithmic framework along with a number of commonly used features (label, syntactic head, etc) to predict function tags with relatively high accuracy. He then presented two other algorithmic frameworks and a number of new features

to be used with them. He proposed to use these expanded systems to improve performance on the function tagging task, and having done so, analyze the results to determine which features were most helpful in the task as a whole and in its various subtasks.

Weiwei Sun and Zhifang Sui [8] addressed the question of assigning function tags to parsed sentences in Chinese. They showed that good performance for Chinese function tagging can be achieved by using labeling method, extending the work of Blaheta (2004). In their method, the objects being modeled are syntax trees which require some mechanism to convert them into feature vectors. They proposed some new features to convert syntax trees into feature vectors. They evaluated on both hand-crafted and automatic parsing syntax trees to clarify the performance of models in Chinese function tag labeling.

Mihai Lintean and Vasile Rus [9] 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 [7] that includes the following: label, parent's label, right sibling label, left sibling label, parent's head POS, head's POS, grandparent's head's POS, parent's head, head. They did not use the alternative head's POS and alternative head (for prepositional phrases that would be the head of the prepositional object) as explicit features but rather modified the phrase head rules so that the same effect is captured in POS and head features, respectively. 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. The results reported are on perfectly parsed trees from the Penn Treebank corpus.

3. MYANMAR LANGUAGE

3.1 Grammar of Myanmar Language

Grammar rules are studied behind languages. The aspect of grammar, which does not concern directly with meaning, 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 only two parts of speech in Myanmar, the noun and the verb, instead of being usually accepted eight parts (Pe Maung Tin 1956:195). Most Myanmar linguists [10] accepted that there are eight parts of speech in Myanmar. Myanmar nouns and verbs need the help of suffixes or particles to show grammatical relation.

For example:
ကျောင်းသူများသာ ဝုဏ်ထူးရသည်။
သူတို့သည် အတန်းထဲမှာ ရှိ**ကြ**၏။

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.

For example:
သူတို့သည် မန္တလေးတွင် ၈ ရက် **တိတိ** လည်ခဲ့သည်။

Stewart remarked that "The Grammar of Myanmar 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.

3.2 Issues of Myanmar Language

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

- In Myanmar sentences, grammatical particles/ postpositional marker (PPM) is mostly used in each phrase and it shows the function/case of the word. Myanmar phrases can be written in any order as long as the verb phrase is at the end of the sentence.
  For example:
  မောင်လှ သည် စာအုပ်တစ်အုပ် ကို မောင်ဘ အား ပေးသည်။
  Ma Hla thi(PPM) sar oat ta oat ko(PPM) Mg Ba arr(PPM) pay thi.
  (or)
  စာအုပ်တစ်အုပ် ကို မောင်ဘ အား မောင်လှ က ပေးသည်။
  Sar oat ta oat ko(PPM) Mg Ba arr(PPM) Ma Hla ka(PPM) pay thi.
  (Ma Hla gives a book to Mg Ba.)
- The phrase order of Myanmar language is free. The sentence can be constructed by placing emphatic phrases at the beginning of the sentence.
  For example:
  သူသည်- သတင်းစာကို - ဖတ်သည်။(Subj-Obj-Verb)
  He - newspaper - reads
  (or)
  သတင်းစာကို - သူ - ဖတ်သည်။ (Obj-Subj-Verb)
  newspaper - he - reads
  (He reads the newspaper.)
- The subject or object of the sentence can be skipped, and still be a valid sentence.
  For example:
  ရန်ကုန်သို့သွားသည်။ (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:
  မောင်မောင် - များ - ပထမ - ဆု - ရ -
  Mg Mg - particle - first - prize - wins -
  လျှင် - သူ့မိဘများ - က - အံ့ဩြ - လိမ့်မည်။
  if - his parents - PPM - surprise - will
  (If Mg Mg wins the first prize, his parents will surprise.)
- In Myanmar language, an adjective can be specialized before or after a noun unlike other languages.
  For example:
  သူသည် - ချမ်းသာသော - လူ -တစ်ယောက် - ဖြစ်သည်။
  He - rich - man - a - is
  (or)

သူသည် - လူ - ချမ်းသာ - တစ်ယောက် - ဖြစ်သည်။
He - man - rich - a - is
(He is a rich man.)

- The subject /object can be another sentence, which does not contain subject or object.
  For example:
  ကလေးများသစ်ပင်အောက်တွင်ကစားနေသည် ကို ကျွန်တော်မြင်သည်။
  (I see the children playing under the tree.)
- The postpositions of subject phrases or object phrases can be hidden.
  For example:
  သူသည်- ဆရာဝန် -တစ်ယောက် - ဖြစ်သည်။
  He - doctor - a - is
  (or)
  သူ - ဆရာဝန် - တစ်ယောက် - ဖြစ်သည်။
  He - doctor - a - is
  (He is a doctor.)
- The postpositions of time phrases or place phrases can be omitted.
  For example:
  သူမ - ကျောင်း - သို့ - သွားသည်။
  She - school - to - goes
  (or)
  သူမ - ကျောင်း - သွားသည်။
  She - school - goes
  (She goes to school.)
- The verb phrase can be hidden in a Myanmar sentence.
  For example:
  သူ - မောင်လှ -ပါ။
  He - Mg Hla - particle
  (He is Mg Hla.)

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

3.3 Syntactic Structure of Myanmar Language

It is known that many postpositions can be used in a Myanmar sentence. If the words are misplaced in a sentence, this sentence can be abnormal one [11]. There are two kinds of sentence according to the sentence construction: 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 the sentence's final particle (sfp) at the end of the sentence [12].

SS=IC+sfp
CS=DC...+IC+sfp
IC may be noun phrase or verb or combination of both.
IC=N... (အိမ်+က+လူ+တွေ) (noun+particle+noun+particle)
IC=V (သွား) (verb)
IC=N...+V(ဘုရား+မှာ+ပန်း+နဲ့+ဆီမီး+လှူ)
(noun+particle+noun+conjunction+noun+verb)
DC is the same as IC but it must contain a clause marker (cm) at the end.
DC=N...+cm (ကျောင်း+က+ဆရာ+ပဲ) (noun+particle+noun+cm)
DC=V+cm (ရောက်+ရင်) (verb+cm)
DC=N...+V+cm (စိတ်ထား+ဖြူ+မှ) (noun+verb+cm)

4. PROPOSED FUNCTION TAGSET

Function tagging is a process of assigning syntactic categories like subject, object, time and place 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.

According to the contextual and functional structure, natural languages are different from each other. Therefore, it is necessary to have a function tagset for Myanmar language.

In English Penn Treebank, there are 20 function tags. These tags are categorized into four groups such as Grammatical, Form/Function, Topicalisation and Miscellaneous. In Chinese Penn Treebank (CTB), there are 26 function tags. These tags are categorized into five groups. They are Syntactic Label, Semantic Label, Miscellaneous Label, Clause Type and Discrepancy Label.

We propose a set of function tags based on the inflecting system and address the question of assigning function tags to Myanmar words. According to Myanmar grammar, there are 17 kinds of postpositional marker (PPM). The function tags are mostly identified with word and postpositional marker (PPM) combination. The function tagset contains 39 function tags. There are one tag for verb phrase and 38 tags for other phrases.

4.1 Function Tag for Verb Chunk

4.1.1 Active/Verb

In our language, the verb phrase must be at the end of the sentence.

For example:
သူ ပြေး(**Active**)သည်။
He runs.

4.2 Function Tags for Other Chunks

4.2.1 Subj, PSubj , SubjP |Subject

The subject of every sentence node gets this tag. Subj tag may be placed at the beginning of the sentence. Besides, it can be placed before verb phrase. PSubj and SubjP tags are combined to produce a Subj tag. The postpositions of Subject phrase are (သည်၊က၊မှာ).

For example:
သူ (**PSubj**) သည် **(SubjP)** ဆရာတစ်ယောက် ဖြစ်သည်။
He is a teacher.
သူ (**Subj**) စျေးသို့ သွားသည်။
He goes to the market.

4.2.2 Obj,PObj,ObjP| Direct Object

These tags mark direct objects in their postpositional form. Obj tag may be placed after the indirect object. PObj and ObjP tags are combined to produce an Obj tag. The postposition of Direct Object phrase is (ကို).

For example:
သူသည် ကော်ဖီ (**PObj**) ကို (**ObjP**) သောက်သည်။
He drinks **coffee**.
သူ ကော်ဖီ (**Obj**) သောက်သည်။
He drinks **coffee**.

4.2.3 PIobj,IobjP|Indirect Object

These tags mark indirect objects in their postpositional form. Two tags are required to form an indirect object. In order to occur these tags, the verb needs to allow the indirect object to appear before the direct object. The postposition of Indirect Object phrase is (အား).

For example:
သူသည် မလှ (**PIobj**) အား (**IobjP**) စာအုပ်ကို ပေးသည်။
He gives a book **to Ma Hla**.

4.2.4 Pla,PPla,PlaP|Place

These tags are used to mark phrases that denote the place where something takes place. Some postpositions of Place phrase are (သို့မှာ၊တွင်၊ဝယ်၊က).

For example:
သူ ကျောင်း (**PPla**) သို့ (**PlaP**) သွားသည်။
He goes **to school**.
သူ ကျောင်း (**Pla**) သွားသည်။
He goes **to school**.

4.2.5 Tim,PTim,TimP|Time

These tags mark temporal constituents, those which answer the questions "when?", "how often?", and "how long?" Time phrases can be either noun phrases or postpositional phrases. Some postpositions of Time phrase are (မှာ၊တွင်၊ဝယ်၊က၊အထိ၊တုန်းက).

For example:
သူသည် နံနက်(၆)နာရီ (**PTim**) တွင် (**TimP**) အိပ်ရာမှထသည်။
He gets up from bed **at 6 o'clock in the morning**.
အမေ နံနက် (**Tim**) စောစော ဈေးသွားသည်။
My mother goes to the market **in the early morning**.

4.2.6 PExt,ExtP|Extract

These tags are usually written with superlative degree adjective. It is used to extract a strange person or thing from the group. The postpositions of Extract phrase are (ထဲမှာ၊တွင်၊အနက်).

For example:
ကျောင်းသားများ(**PExt**) အနက်(**ExtP**) မောင်ဘသည် အ တော် ဆုံး ဖြစ်သည်။
Mg Ba is the cleverest boy **among students**.

4.2.7 PSim,SimP|Simile

It is used to show a work of fantasy. Some postpositions of Simile phrase are (ကဲ့သို့ ၊လို၊ပမာ).

For example:
သူမသည် မင်းသမီး(**PSim**) လို(**SimP**) ဝတ်စား၏။
She wears the dress **as an actress**.

4.2.8 PCom,ComP|Compare

Sometimes, it is written with adverb such as အတူ (together). The postposition of Compare phrase is (နှင့်အတူ).

For example:
သူသည် သူဦးလေး (**PCom**) နှင့်အတူ(**ComP**) နေသည်။
He lives **with his uncle**.

4.2.9 POwn,OwnP|Own

These tags concern with the things of owner so it is accepted as adjective. This postposition describes the owner. The postpositions of Own phrase are (၏ ၊ရဲ့).

For example:
သူသည် သူ (**POwn**) ၏ (**OwnP**) အမေကို ချစ်သည်။
He loves **his mother**.

4.2.10 Ada|Predicative Complement

In our language, an Adjective phrase can be formed by combining an adjective and the sentence's final particle.

For example:
သူမ လှ (**Ada**) သည်။
She is **beautiful**.

4.2.11 PcomplS|Subject Complement

Although a sentence contains subject and verb, it is not the meaningful sentence. So, a phrase can be added after Subject phrase. This phrase is called subject complement.

For example:
သူမသည် ဆရာမ (**PcomplS**) ဖြစ်သည်။
She is **a teacher**.

4.2.12 PcomplO,PPcomplO,PcomplOP|Object Complement

Although a sentence contains subject, object and verb, it is not the meaningful sentence. So, a noun phrase can be added after object phrase. This phrase is called object complement.

For example:
မောင်လှသည် ရွှေကို လက်စွပ် (**PcomplO**) လုပ်သည်။
Mg Hla makes the gold **a ring.**
သူတို့သည် ဦးဘကို ခေါင်းဆောင်(**PPcomplO**) အဖြစ်(**PcomplOP**) ရွေးချယ်ခဲ့ သည်။
They selected U Ba **as a leader**.

4.2.13 PUse,UseP|Use

The Use phrase that contains (နှင့် ၊ ဖြင့်) postpositions can usually be placed before a verb phrase to get a meaningful sentence.

For example:
သူသည်ခွေးကို တုတ်(**PUse**)ဖြင့်(**UseP**) ရိုက်သည်။
He hits the dog **with a stick**.

4.2.14 PCau,CauP|Cause

These tags are used to describe the cause of an event. The postpositions of Cause phrase are (ကြောင့်၊အတွက်ကြောင့်).

For example:
လယ်ကွင်းများသည် မုန်တိုင်း(**PCau**) ကြောင့်(**CauP**) ပျက်စီး သည်။
The fields are destroyed **because of the storm.**

4.2.15 PAim,AimP|Aim

These tags mark constituents that annotate an action of the purpose or the reason. The postpositions of Aim phrase include include (အတွက် ၊ဖို့ ၊အလို့ငှာ).

For example:
သူသည် သူအမေ (**PAim**) အတွက် (**AimP**) ကိတ်မုန့် ဝယ် သည်။
He buys the cake **for his mother.**

4.2.16 CCC,CCS,CCM,CCA,CCP|Conjunction

CCC tag joins the phrases. CCS tag joins the sentences. CCM tag joins the meanings that are included in the independent clauses. CCA tag joins the sentences as an adjective. CCP tag is used to join the sentences as a particle.

For example:

မမ နှင့်**(CCC)**လှလှ သည် သူငယ်ချင်းများ ဖြစ်ကြသည်။
Ma Ma **and** Hla Hla are friends.
မိုးရွာ လျှင်**(CCS)** သူ ဈေး မသွားပါ။
If it rains, he will not go to the market.
သူစာကြိုးစားသည်။ **ထို့ကြောင့်(CCM)** သူ စာမေးပွဲ အောင်သည်။
He tries hard. **So**, he passes the examination.
အဖေ ဖယ်လာ **သော (CCA)**စာအုပ်ကို ကျွန်တော် ဖတ်နေသည်။
I am reading the book **that** my father bought.
သူဘောလုံးကစားနေသည် **ကို(CCP)** ကျွန်တော် မြင်ခဲ့သည်။
I saw **that** he is playing the football.

5. CORPUS CREATION

We collected several types of Myanmar texts to construct a functional annotated tagged corpus. Our corpus is to be built manually. We extend the POS tagged corpus that is proposed in [1]. The chunk and function tags are manually added to the POS tagged corpus. The corpus contains about 2150 sentences with average word length 15. All sentences are collected from Myanmar textbooks of middle school and Myanmar grammar books. They are simple sentences and complex sentences. Manually annotated corpora are valuable but scarce resources. The corpus data will be annotated only up to the sentence level in order to be in the same format for all Myanmar language. The corpus size is bigger and bigger because the tested sentences are automatically added to the corpus. Myanmar books and websites are text collections (see Table 1).

Table 1: Corpus statistics

Text types	# of sentences
Myanmar textbooks of middle school	742
Myanmar Grammar books	683
Myanmar websites	440
Others	285
Total	2150

In our corpus, a sentence contains chunk, function tag, word and its POS tag with category (see Figure 1).

NC@PSubj[သူမ/pron.person] # PPC@SubjP[သည်/ppm.subj] # NC@PObj [လူနာ/n.person,များ/part.number] # PPC@ObjP[ကို/ppm.obj] # NC@PSim [ဆွေမျိုး/n.person,များ/part.number] # PPC@SimP[ကဲ့သို့/ppm.sim] # VC@ Active[ပြစု/v.common] # SFC@Null[သည်/sf]။

Figure 1: A sample sentence in the corpus

6. NAÏVE BAYESIAN 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 naive Bayesian classifier involves learning how much each function tag should be trusted for the decisions it makes. In probability estimation for Naive Bayesian classifiers, namely that the attribute values are conditionally independent when the target value is given. Naive Bayesian classifiers are 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.

6.1 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 describing a sentence. 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 in 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, 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 (see Table 2).

Table 2: Features

Feature	English	Myanmar
n.animals	dog	ခွေး
pron.person	he	သူ
ppm.place	in	တွင်
cc.sentence	if	တွင်

In Myanmar language, some words have the same meaning but in different features (see Table 3).

Table 3: Same word with different features

Feature	English	Myanmar
cc.chunk	and	နှင့်
ppm.compare	with	နှင့်

A class is one of the proposed function tags. The same word may have different function tags (see Table 4).

Table 4: Function tags

Function tags	English	Myanmar
Subj	The **book** is on the table.	စာအုပ်
Obj	He left the **book** at school.	စာအုပ်

6.1.1 Training

There are many chunks in a sentence such as NC (noun chunk), PPC (postpositional chunk), AC (adjective chunk), RC (adverbial chunk), CC (conjunctional chunk), SFC (sentence final chunk) and VC (verb chunk). A chunk contains a Myanmar head word and its modifier. It may have 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 [ခွေး/n.animals,တစ်/part.number,ကောင်/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 [ဆောင်းရာသီ/n.time,သီးနှံပင်/n.food,များ/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 (see Table 5).

$$\mathbf{P(t_k|pc) = C(t_k,pc)/C(pc)} \quad (1)$$

We calculate the probability between next function tags (n_1, $n_2 \ldots n_j$) and previous possible tags by using the following equation that is log likelihood (see Table 6).

$$\mathbf{P(n_j|t_k) = C(n_j,t_k)/C(t_k)} \quad (2)$$

6.1.2 Disambiguation

Possible function tags are disambiguated by using Naïve Bayesian method. We multiply the probabilities from (1) and (2) 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.

w	a word
c	category of a word
pc	POS tag word with category
$t_1, t_2 \ldots t_k$	possible tags of the word with category
$n_1, n_2 \ldots n_j$	possible tags of the next word with category
C (t_k,c)	the number of occurrences of t_k followed by c
C(c)	the number of occurrences of c in the training set
C (n_j,t_k)	the number of occurrences of n_j followed by t_k
C (t_k)	the number of occurrences of t_k in the training set

Figure 2: Notational conventions for function tagging

comment: Training

for a pc of w **do**
  for all tags t_k of pc **do**
    P (t_k|pc) = C(t_k,pc)/C(pc)
  end
end

for all tags t_k of wc **do**
  for all tags n_j of wc **do**
    P (n_j|t_k) = C(n_j,t_k)/C(t_k)
  end
end

comment: Disambiguation

for all tags t_k of wc **do**
  score (t_k)=log P(t_k)
  for all tags n_j in the next wc **do**
    score (t_k)=score(t_k)+log P(n_j|t_k)
  end
end

choose t=arg max t_k score (t_k)

Figure 3: Naïve Bayesian classification for function tags disambiguation

Our description of the function tagging process refers to the example (see Figure 4), which illustrates the sentence "I read the book that is given by my father" ("အဖေပေးသောစာအုပ်ကိုကျွန်တော် ဖတ်သည်။").

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".

(a) NC [အဖေ/n.person] # VC [ပေး/v.common] # CC [သော/cc.adj] # NC [စာအုပ်/n.objects] # PPC [ကို/ppm.obj] # NC [ကျွန်တော်/pron.person] # VC [ဖတ်/v.common] # SFC [သည်/sf]။

(b)Subj[အဖေ] # Active[ပေး] # CCA[သော] # PObj[စာအုပ်]# ObjP[ကို] # Subj [ကျွန်တော်] # Active[ဖတ်သည်]။

Figure 4: An overview of the function tagging of the sentence (a) The input tagged and chunk sentence (b) The output sentence with function tags

Table 5: Idealized probabilities for the tags that some words occur with in the corpus For example, 40% occurrences of n.animals are with the tag Subj

POS tags	Function tags											
	Subj	**Obj**	**PTim**	**Tim**	**Pla**	**PcomplO**	**PUse**	**PSim**	**SimP**	**CCS**	**Ada**	**Active**
n.animals	0.4	0.17	0	0	0	0.08	0.07	0.28	0	0	0	0
n.objects	0.31	0.1	0	0	0.05	0.02	0.17	0	0	0	0	0
cc.sent	0	0	0	0	0	0	0	0	0	1	0	0
ppm.sim	0	0	0	0	0	0	0	0	1	0	0	0
pron.person	0.48	0.02	0	0	0	0	0	0.19	0	0	0	0
n.location	0.21	0.04	0	0	0.65	0.04	0	0.01	0	0	0	0
v.common	0	0	0	0	0	0	0	0	0	0	0	1
adj.dem	0	0	0	0	0	0	0	0	0	0	0.87	0.13
n.time	0	0	0.58	0.42	0	0	0	0	0	0	0	0

Table 6: Idealized probabilities of some tag transitions in the corpus. For example, PExt occurs 8% after Subj

	Second Function tags

First Function tags	Subj	Obj	PTim	TimP	Pla	PExt	PAim	PCau	CauP	CCC	CCS	Active
Subj	0	0.07	0.01	0.01	0.01	0.08	0.03	0	0	0.03	0.01	0.68
Obj	0.02	0	0	0	0	0	0	0	0	0.05	0	0.89
PTim	0	0	0	0.9	0	0	0	0	0	0.1	0	0
TimP	0.25	0.14	0	0	0.06	0	0	0.08	0	0	0	0.36
Pla	0.11	0.02	0.06	0	0	0.02	0.01	0.04	0	0	0	0.84
PExt	0	0	0	0	0	0	0	0	0	0	0	0
PAim	0	0	0	0	0	0	0	0	0	0	0	0
PCau	0	0	0	0	0	0	0	0	0.84	0.16	0	0
CauP	0.43	0.01	0.06	0	0.28	0.13	0.07	0	0	0	0	0.15
CCC	0.49	0.2	0.04	0	0.08	0.1	0.04	0	0	0	0	0
CCS	0.25	0.19	0.02	0	0.06	0.01	0.01	0	0	0	0	0.33
Active	0	0	0	0	0	0	0	0	0	0	0.27	0

7. EVALUATION METHOD

For evaluation purpose, different numbers of sentences collected from Myanmar textbooks and Myanmar websites are used as a test set. In our test set, 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. In complex sentences, they can be further classified as three groups. They are sentences which are combined by 2 clauses (DC+IC), 3 clauses (DC+DC+IC) and 4 clauses (DC+DC+DC+IC). Therefore, we will obtain complete knowledge about the performance of the function tagging by the comparison of it on these two types of sentences. There are 60 sentences in the first group and 90 in the second one.

The precision and recall of function tagging is measured for different sentence types and different sentence structures since the complexity of function tagging mostly relies on sentence types and patterns. In measuring precision and recall, precision and recall for overall function tagging is calculated. For the context of function tagging, precision is the ratio of number of function tags which are correctly tagged to the total function tags. Recall is the ratio of function tags which are correctly tagged to the number of actual existing function tags. The precision and recall of function tagging is calculated by using (3) and (4) respectively.

$$\text{Precision} = \frac{NumberOfCorrectFunctionTags}{TotalFunctionTags} \times 100\% \quad (3)$$

$$\text{Recall} = \frac{NumberOfCorrectFunctionTags}{NumberOfActualExistingFunctionTags} \times 100\% \quad (4)$$

We found that the lack of postpositional makers in the input sentences affect on the performance. As we increase the number of postpositional markers that are used in the evaluation, the precision increased. The accuracy of complex sentences is higher than the simple sentences because clauses in most of the complex sentences form simpler and shorter length than a complete simple sentence. We show the performance of function tagging according to the two groups (see Table 7 and Figure 5).

Table 7: Performance of function tagging for different sentence types

Sentence Types	Precision (%)	Recall (%)
Simple	94.31	90.58
Complex Sentences with 2 clauses	92.58	89.02
Complex Sentences with 3 clauses	95.34	91.21
Complex Sentences with 4 clauses	96.76	93.74

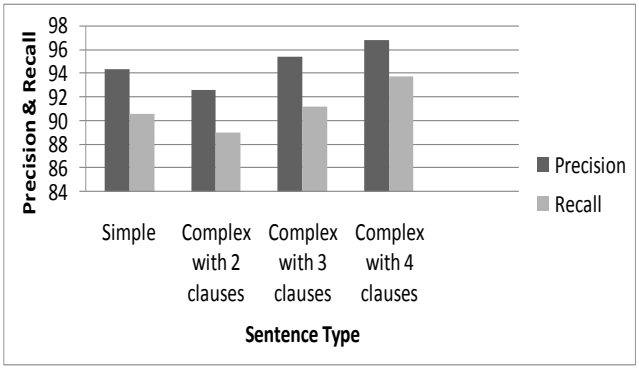

Figure 5: Result of function tagging on testing set

8. ERROR ANALYSIS

We conclude our remarks on tagging accuracy by giving examples of some of the most frequent errors. We show some examples of common error types reported (see Table 8). The example phrases and fragments are all ambiguous, demonstrating that semantic context, or more syntactic context is necessary than a Bayes Model has access to. The most common mistakes occur between Subj and PcomplS. Both tags are especially placed before verb phrase. It is noticed that in over 50% of cases when a subject complement (PcomplS) was misidentified as Subject (Subj). Syntactically, the word (ဆရာမ) could be a Subj as well as a PcomplS. The word (အိမ်) could refer to the subject complement. One typical error is caused by the lack of postpositional markers in a sentence. For the word (အတန်း), Subj and Obj are sometimes mistagged when the

sentences omit the PPM. The word (ကျောင်း) could refer to the PSubj tag. Finally, depending on the embedding PPM, (a book) can be a PcomplS or an Obj form as the following two sentences show:

a.စားပွဲ ပေါ်တွင် စာအုပ်တစ်အုပ် ရှိသည်။
b.စားပွဲ ပေါ်တွင် စာအုပ်တစ်အုပ် တွေ့သည်။

Table 8: Examples of frequent errors of probabilistic function taggers

Correct tag	Tagging error	Example
PcomplS	Subj	မမ သည် ဆရာမ ဖြစ်သည်။
PcomplS	Pla	သူမှာ အိမ် ရှိသည်။
Obj	Subj	အတန်း မပျက်ကွက်ပါ။
PSubj	Pla	ကျောင်းနှင့်အိမ်သည် နီးသည်။
PcomplS	Obj	စာအုပ်တစ်အုပ်
Obj	PcomplS	စာအုပ်တစ်အုပ်

9. ACKNOWLEDGMENTS

I am deeply indebted to my supervisor Prof. Dr. Ni Lar Thein, Rector, University of Computer Studies, Yangon and my co-supervisor Associate Prof. Dr. Tin Myat Htwe, Computer Software Department, University of Computer Studies, Yangon whose help, stimulating suggestions and encouragement helped me in all the time of research.

10. CONCLUSION

In this paper, 39 function tags for Myanmar language is proposed and the function tag of the word is investigated depending on the sentence structures of Myanmar language. Naïve Bayesian technique is used to disambiguate the task of assigning function tags. The corpus is the resource for the development of Myanmar to English translation system and it is expected that the size of functional annotated tagged corpus is bigger in the future because the tested sentences can be added into the corpus. Function tags have in the past not been very well studied or exploited. Because of the lack of prior research on this task, it is impossible to compare our results to those of other researchers; but the results do seem promising.

And this research is an ongoing stage of developing Myanmar to English machine translation system. Several challenges are expected to encounter within near future.

11. REFERENCES

[1] Myint, P. H. Assigning automatically Part-of-Speech tags to build tagged corpus for Myanmar language, The Fifth Conference on Parallel Soft Computing, Yangon, Myanmar, 2010.

[2] Myint, P. H., Chunk Tagged Corpus Creation for Myanmar Language. In Proceedings of the ninth International Conference on Computer Applications, Yangon, Myanmar, 2011.

[3] Charniak, E. 1999. A maximum-entropy inspired parser. Technical Report CS-99-12, Brown University, August.

[4] Charniak, E. 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.

[5] Versteegen, L. 1999. The Simple Bayesian Classifier as a Classification Algorithm.

[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 forComputational Linguistics, 234–240.

[7] Blaheta, D. 2003. Function tagging. Ph.D. Dissertation, Brown University. Advisor-Eugene Charniak.

[8] Sun, W., and Sui, Z. Chinese Function Tag Labeling. In Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC 2009*)*, Hong Kong, 2009.

[9] Lintean, M., and Rus, V. 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 2007 (in press).

[10] Soe, S. P. 2010. Aspects of Myanmar Language , Myanmar Department, University of Foreign Language.

[11] Myanmar Thudda, vol. 1 to 5 in Bur-Myan, Text-book Committee, Basic Edu., Min. of Edu., Myanmar, ca. 1986.

[12] Lay, K. 2003. Construction of Myanmar Thudda. Ph.D. Dissertation, Myanmar Department, University of Educaion.