MPCS: Myanmar Preposition Checking System

Khaing Htet Win
University of Computer Studies, Yangon

ABSTRACT
For every language, preposition checker is essential component of many of Office Automation System and Machine Translation System. In addition, Myanmar prepositions play an important role of Myanmar sentences because the percentages of preposition errors are the highest in Myanmar sentence. This paper describes a Transformation Based Learning (TBL) Algorithm to the automatic correction of preposition errors in Myanmar Language. TBL uses rule templates to identify error-correcting patterns. A critical requirement in TBL is the availability of a problem domain expert to build these rule templates. In this work, Decision tree (DT) is used to automatically generate TBL rule templates. Myanmar Preposition Checking System (MPCS) which can handle missing preposition errors, misused preposition errors and unwanted preposition errors. As a Resource, a Myanmar Text Corpus is created and Myanmar3 Unicode is applied in this system. This proposed system improves the quality of corrections for Myanmar prepositions errors in students and non-native writers. It also provides the quality of machine translation system and many NLP applications.

General Terms
Natural Language Processing (NLP)

Keywords
Myanmar Prepositions Checking System (MPCS), Transformation based learning (TBL) Algorithm, Decision Tree (DT)

1. INTRODUCTION
Myanmar language is similar to other Asian Language including Indian, Japanese, Thai and Chinese Language. In our country, Myanmar Language is used as an official language so preposition checker is an essential role for the development of NLP. In addition, prepositions are challenging for learners because they can appear to have an idiosyncratic behavior which does not follow any predictable pattern even across nearly identical contexts. That is, the choice of a preposition for a given context also depends upon the intention of the writer. For example, if preposition errors contain in the input sentences, these sentences may not be meaningful or change the meaning of sentences.

For example,

ဆရာမ သည် မမ အား /ကို စာအုပ် ကို /အား ပေးသည်။ (The Teacher gives a book Ma Ma.)

In this sentence, the writer can confuse the placement of prepositions (အား/ကို). One approach to solving this problem would be to rely on our linguistic intuitions to manually generate a set of rules. However, this can be time consuming. Since the rules of language are vast and idiosyncratic, a person would likely miss important and powerful rules if relying solely on intuition.

One recently proposed approach [5] for rule learning is transformation-based learning. Transformation based learning has been applied to a number of natural language problems, including part of speech tagging, prepositional phrase attachment disambiguation speech generation and syntactic parsing often achieving state-of-the-art accuracy while capturing the learned knowledge in a small and easily understood set of rules.

In [3], a randomized version of the TBL framework is shown. The idea is to use just a few templates, randomly chosen from the template set, when generating candidate rules for each error. This strategy speeds up the TBL training process enabling the use of large template sets. On the other hand, in the experiments on Part-of-speech tagging, Carberry et al use handcrafted templates and variations of them, what implies that a template designer is still necessary.

To overcome this problem, the combination of Decision Trees (DT) and Transformation Based Learning (TBL) is applied to this system. The combination of DT and TBL is a new machine learning strategy. The key idea is to use decision tree induction to obtain templates. Next, the TBL strategy is used to generate transformation rule. This combination is more effective than only decision trees and also eliminates the need of a problem domain expert to build TBL templates.

The rest of this paper is organized as follows: Section 2 describes the related works. Section 3 describes features of Myanmar language. Section 4 presents the types of preposition errors. Section 5 presents a brief overview of TBL. Section6 is depicted the framework of Myanmar preposition checker. Section 7 describes experimental results and conclusion is given in section 8.

2. RELATED WORKS
Many researchers have been worked for preposition checker of Asian Languages. Even though other Asian preposition checker researches have been done for two decades, Myanmar Preposition Checker research is not still developed as I am concerned. And then, there is a very little amount of work done in natural language processing of Myanmar Language. In this section, some of the related work and history are briefly discussed in the area of preposition checking systems.

Most of the methods for correcting preposition error are based on supervised approaches. An unsupervised method for correcting preposition errors in French as a second language is presented in [2] and it uses counts collected from the Web in a simple way, in order to rank the candidates.

Eeg-Olofsson [6] used 31 handcrafted matching rules to detect extraneous, omitted, and incorrect prepositions in Swedish text written by native speakers of English, Arabic, and Japanese. In a test of the system, 11of 40 preposition errors were correctly detected.

Carberry et al. [7] introduce a randomized version of the TBL framework. For each error, they try just a few randomly chosen templates from the given template set. This strategy speeds up the TBL training process, enabling the use of large template sets. However, they use handcrafted templates and variations of them, which imply that a template designer is still necessary.

An evolutionary approach based on Genetic Algorithm (GA) to automatically generate TBL templates is presented in [8]. Using a simple genetic coding, the generated template sets have efficacy near to the handcrafted templates for the task: English Base Noun Phrase Identification, Text Chunking and Portuguese Named Entities Recognition. The main drawback of this strategy is that the GA step is computationally expensive.

In contrast to previous research, a preposition grammar checker is implemented by Transformation Based Learning (TBL) and Decision Trees (DT).Combination of these two methods can effectively reduce preposition error and improve the correctness of sentences.

3. MYANMAR LANGUAGE FEATURES

Myanmar Language belongs to Tibeto-Burman language family and derives from Sino-Tibetan language tree. Myanmar Script has been a majority language of Myanmar over 1000 year old. Myanmar Language Commission defined that, in Myanmar thinbongyi (primer), there are 33 consonants: beginning with □ and ending with □. There are nine Part-of-Speech classes for all Myanmar words since it is described by Myanmar Language Commission [13-14]. These are Noun ("နာမ်"), Pronoun ("နာမ်စား"), Verb ("ကြိယာ"), Adjective ("နာမဝိသေသန"), Adverb ("ကြိယာဝိသေသန"), Conjunction ("သမ္မန္ဓ "), Postpositional ("ဝိဘတ် "), Particles ("ပစ္စည်း ") and Interjection ("အာမေဋိတ် ").

Preposition class in English is mostly the same with postpositional in Myanmar. In many languages like Myanmar, Urdu, Turkish, Hindi and Japanese, the words with this grammatical function come after, not before, the complement. Such words are then commonly called postpositions. Sample postpositions in Myanmar Language are "သို့ " [-thou.], " ကို " [-kou], " သည် " [-thi] .

These prepositions are divided into two parts: prepositions for noun which support the nouns in behind and prepositions for verb which support the verb in behind. Among them, preposition for noun which are divided as following Table 1. Each category of prepositions is also divided into various parts according to their features. Although three prepositions သည်၊ က၊ မှာ [-thi, ka, hma] involves in same categories, each word emphasized on different meaning of sentence. These confusions of prepositions lead to the most sentence errors.

Table1. The Category of Myanmar Noun Prepositions

No	Category	Preposition
1	Subject	သည်၊
2	Subject	က
3	Subject	မှာ
4	Object	ကို
5	Accept	အား
6	Direction	သို့
7	Used	ဖြင့်၊ နှင့်၊ နဲ့
8	Leave	မှ၊က
9	Time	၌၊ မှာ၊ တွင်၊ ဝယ်၊ က
10	Place	၌၊ မှာ၊ တွင်၊ ဝယ်၊ က
11	Time Start	က၊ တုန်းက၊ ကမှ၊ထဲက
12	TimeStart	မှ၊ ကျမှ၊ ကျရင်
13	TimeStart	က၊ထဲက၊ကတည်းက၊တုန်းက၊တည်းက
14	TimeEnd	အထိ၊ အရောက်၊ တိုင်အောင်
15	TimeStart-TimeEnd	မှ၊က၊ကနေ၊မှသည်၊...သို့၊အထိ၊ထိအောင်
16	ContinuousTime	ပတ်လုံး၊ တိုင်တိုင်၊ကြာ
17	Cause	ကြောင့်၊နှင့်၊အတွက်ကြောင့်
18	Extract	ထဲမှာ၊ အနက်၊ တွင်
19	Aim	အတွက်၊ ဖို့၊ အတိုင်း၊ အရာအလို့ငှါ
20	Slimile	ကဲ့သို့၊လို၊ သဖွယ်၊ပမာ၊ အတိုင်း၊ အလား
21	Compare	နှင့်၊ နဲ့၊ အတူ၊ အတူတကွ
22	ContinuousPlace	တလျောက်
23	Possessive	၏၊ ့-

4. TYPES OF PREPOSITION ERRORS

Preposition error can result generally from the mistake made by human. Generally, human-generated errors can be distinguished into three groups:

(1) Missing Preposition

(2) Misused Preposition

(3) Unwanted Preposition

Missing Preposition: Missing errors have been made by the typist accidentally forget to press the prepositions. These errors are made assuming that the writer or typist knows how to use preposition but he/she forget to type these words. For example, ခွေးသည် ကြောင် ကိုက်သည်။ (The dog bite the cat). In this sentence, the error can be occurred between ကြောင် (cat) and ကိုက် (bite). The typist needed to type the preposition ကို [-kou] which supports the object of sentence (ကြောင်). If the writer forgets to write noun preposition ကို [-kou], the sentence is no meaning. The correct sentence is ခွေးသည် ကြောင် ကို ကိုက်သည်။ (The dog bite the cat).

Misused Preposition: Misused error is occurred when they have been made by a lack of knowledge of the writher or typist (e.g., က [ka] as မှ [hma]). These errors are made when the writer substituted letters they confuse in prepositions.

Example:

(i) မြန်မာစာ အဖွဲ့ မှ ထုတ်ဝေ သည်။

(ii) သဘာပတိကြီး မှ မိန့်ခွန်း ပြောကြား ပါလိမ့် မည်။

(iii) ကျောင်းအုပ်ကြီး မှ ဆု ချီးမြှင့်သည်။

There has misused prepositions in three sentences. Some writer misused the preposition like that the word (မှ) is used. The correct preposition for these sentences is (က). In Myanmar words, (မှ and က) are confuse in preposition but difference meanings and difference usages.

Unwanted Preposition: These errors can be caused in writing Myanmar sentences. These prepositions (e.g., အနက် [a-nat] and တွင် [twin]) are words that extract one thing from a group or collection. Therefore, only preposition (e.g., □□□ or □□□) can be used in order to extract one thing from the group. Both prepositions should not be used in one sentence.

For example: "မိန်းကလေးများ အနက် တွင် မယ်မြ သည် အချောဆုံး ဖြစ်သည်" (Ma Mya is the most beautiful among girls). In this sentence, errors are made when the writer used extra preposition. In Myanmar Grammar there is no combination of these two propositions (အနက် and တွင်). They can be used to separately as follow:

မိန်းကလေးများ အနက် မယ်မြ သည် အချောဆုံး ဖြစ်သည်
(or)
မိန်းကလေးများ တွင် မယ်မြ သည် အချောဆုံး ဖြစ်သည်

5. TRANSFORMATION-BASED LEARNING

Eric Brill [17] proposes Transformation Based Learning (TBL). TBL is a corpus-based, error-driven approach in which a set of transformation rules is learned to correct the errors of a baseline classifier. TBL has been successfully used for several Natural Language Processing tasks, such as part-of-speech tagging, phrase chunking, spelling correction, appositive extraction, named entity recognition and semantic role labeling. As input, TBL requires corpus (the labeled data), a baseline classifier and a set of rule templates.

A set of rule templates determines the space of allowable transformation rules. A rule template has two components: a triggering environment (condition of the rule) and a rewrite rule (action taken). On each iteration, these templates are instantiated with features of the constituents of the templates when the condition of the rule is satisfied.

This process eventually identifies all possible instantiated forms of the templates. Among all these possible rules, the transformation whose application results in the best score—according to some objective function—is identified. This transformation is added to the ordered list of transformation rules.

The learning stops when there is no transformation that improves the current state of the data or a pre specified threshold is reached. When presented with new data, the transformation rules are applied in the order that they were added to the list of transformations. The output of the system is the annotated data after all transformations are applied to the initial annotation.

Some advantages of Transformation based learning include the following: simple conceptually, TBL can be adapted to different learning problems, rich triggers/rules can make use of specific information and context, seemingly resistant to over-fitting (observed empirically, not entirely understood).

6. FRAMEWORK OF MYANMAR PREPOSITION CHECKER

Myanmar Prepositions Checker consists of two modules.

They are:

(1) Template Generation Module

(2) Checking Module

6.1 Template Generation Module

Template Generation Module consists of four components as shown in Figure 1. They are: (1) Myanmar Text Corpus, (2) Learn Decision Tree (3) Decompose DT and Extract Templates and (4) Templates.

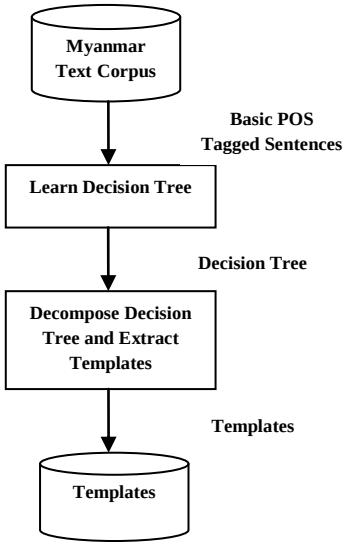

Fig.1 System Flow of Template Generation

Myanmar Text Corpus: This corpus is a large and structured set of texts. Building of the text corpus is very helpful for the development of preposition checking. In this work, Myanmar text corpus is created manually to apply in Myanmar Preposition Checker system. Its various training sentences and all are collected from example sentences of "Myanmar Grammar" [11] and "Myanmar Words Commonly misspelled and misused books [10]".Myanmar Training sentences consists of 2000 sentences and average words in sentences is 12.

Learn Decision Tree: Decision tree learning is one of the most widely used machine learning algorithms. In Template Generation Module, decision tree learning performs a partitioning of Myanmar Text corpus using principles of Information Theory. Information Gain Ratio, which is based in the data Entropy, is normally used as the informativeness measure. This partitioning is defined as

$$H(T) = -\sum_{i=0}^{|C|} P_T(C_i)\log_2 P_T(C_i)$$

Where Ci is a class label from C, |C| is the number of classes and P_T (Ci) is estimated by the percentage of examples belonging to ci in T. In feature selection, information gain can be thought as the expected reduction in entropy H (T) caused by using a given feature A to partition the training examples

in T. The information gain IG (T, A) of a feature A, relative to an example set T is defined as

$$\mathrm{IG(T,A)} = H(T) - \sum_{v \in Value\ (A)}^{n} \frac{|T_V|}{|T|} H(T_V)$$

Where Values (A) is the set of all possible values for feature A, and T_v is the subset of T for which feature A has value v. Next, the learning algorithm executes a depth-first traversal of the DT. For each visited tree node, template is created in the path from root to this node.

Decompose DT and Extract Templates: There are many ways to extract feature combinations from decision trees. In a path from the root to the leaves, more informative features appear first. The process of extracting templates from a DT includes a depth-first traversal of the DT. This feature combination provides an information gain driven template. Additionally, paths from the root to internal nodes also provide good templates.

Templates: Templates are stored in the database and these templates are input into Rule Derivation process of Checking Module.

6.2 Checking Module

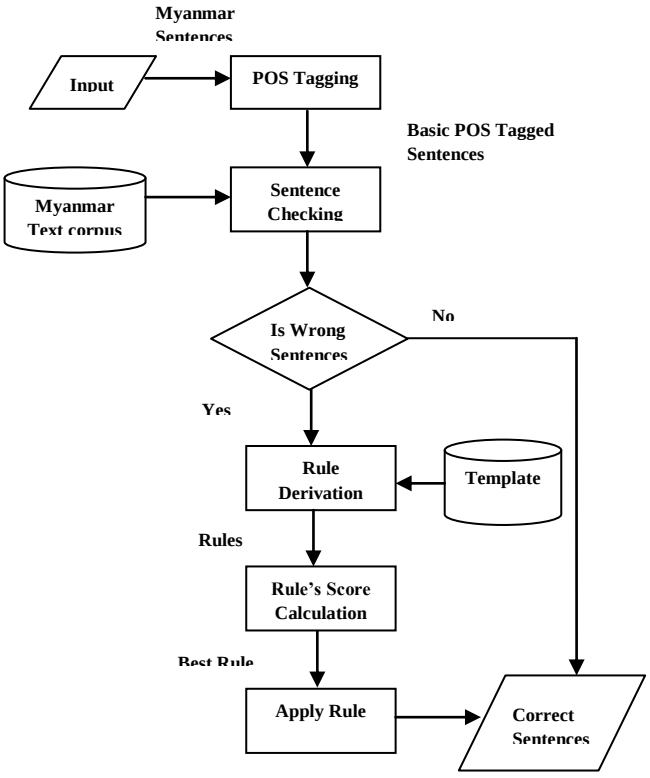

Fig.2 System Flow of Checking Module

To check Myanmar preposition errors, the system uses TBL algorithm as an error correcting strategy. Its main scheme is to generate an ordered list of rules that correct classification mistakes in the training set, which have been produced by an initial guess. The requirements of the algorithm are:

1. Myanmar Text corpus that has been tagged with the basic POS tags.
2. An initial classifier, POS tagging process, which tags the unlabeled sentence by trying to guess the correct basic POS tags for each sample.
3. A set of rule templates, which are meant to capture the relevant feature combinations that would determine the sample's classification. Correct rules are acquired by instantiation of this predefined set of rule templates.

The learning method is a mistake-driven greedy procedure that iteratively acquires a set of transformation rules. The TBL algorithm can be depicted as follows:

1. The input sentence is Myanmar sentence. This sentence is segmented into segments and tagged them with basic POS tag within POS tagging process. After the POS Tagging process, the basic POS tagged sentence is produced as output.
2. Compares POS tagged sentence with Myanmar Text Corpus, whenever an error is found, all the rules that can correct it are generated by instantiating the templates. This template instantiation is done by capturing some contextual data of the sample being corrected. Usually, a new rule will correct some errors, but will also generate some other errors by changing correctly classified samples.
3. Computes the rules scores (errors repaired-errors created). If there is not a rule with a score above an arbitrary threshold, the learning process is stopped.
4. Finally, the best scoring rules is applied to the POS tagged sentences and produce the correct sentence.

6.2.1 Transformation Based Learning Algorithm for checking module

Input: Myanmar sentences; Template Set; POS Tagger; RuleScoreThreshold

1. apply (POS Tagger, Myanmar Sentences) → POS Tagged Sentences
2. repeat
3. Candidate Rule ←{ }
4. for all example Є POS Tagged Sentences do
5. if isWronglySentence (example) then
6. for all template Є Template Set do
7. instantiate Rule (template, example)→rule
8. Candidate Rules ← Candidate Rules + rule
9. end for
10. end if
11. end for
12. bestScore ← 0
13. bestRule ← Null
14. for all rule Є Candidate Rules do
15. count Corrections (rule, POSTaggedSentences)→ good
16. count Errors (rule, POSTaggedSentences) → bad
17. score = good – bad

18. if score > bestScore then
19. bestScore ← score
20. bestRule ← rule
21. end if
22. end for
23. if bestScore > RuleScoreThreshold then
24. Correct Sentences ← apply (BestRule, POS Tagged Sentences)
25. end if
26. until bestScore > RuleScoreThreshold
27. output Correct Sentences

In this pseudo-code, the *apply* function tags the Myanmar sentences using the POS Tagging process. The *isWronglyClassified* function checks whether the sentence is correct or not. This checking is done by comparing the current sentence to the correct Myanmar Text corpus. The *instantiateRule* function creates a new rule by instantiating the given template with the given sentence context values. The *countCorrections* function returns the number of corrections that a given rule would produce in the current training set. Similarly, the *countErrors* function returns the number of errors that a given rule would produce in the current training set. There are also several variants of the TBL algorithm. FastTBL [18] is the most successful, since it achieves a significant speedup in the training time while still achieving the same performance as the standard TBL algorithm.

7. EXPERIMENT RESULTS

This paper emphasizes on the preposition checking that can correct the most error percentages of Myanmar Sentences. Three testing paragraphs are used for evaluation in order to calculate the accuracy of the preposition checker and each paragraph contains 250 sentences. First paragraph A contains 16% preposition errors of the total words in the paragraph. Second paragraph B has 39% preposition errors and third paragraph C has 63% preposition errors.

The performance of this system is evaluated in terms of precision, recall and F-measure. Precision (P) means the percentage of the correct word suggested by the system which is divided by total number of error detected by the system. Recall (R) means the percentage of correct words suggested by the system which is divided by the total number of sentence. F-score is the mean of recall and precision, that is F= 2PR / (P+R). The following figures show the accuracy of correctly detected on the testing sentences with Myanmar Preposition Checker. In these figures, suggestion generation of Average accuracy of overall system gets 95% precision, 92.33% recall and 93% f-score.

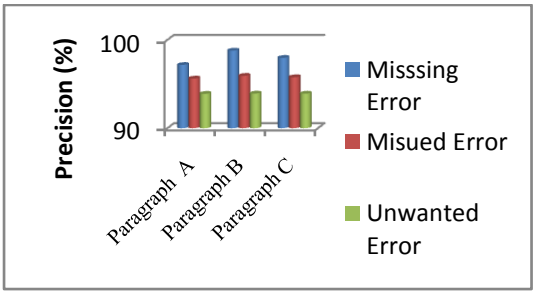

Figure3. Precision result of overall system evaluation

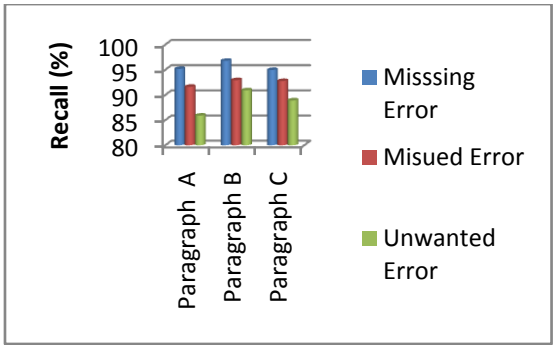

Figure4. Recall result of overall system evaluation

Figure5. F-Score result of overall system evaluation

Table2. Experimental Results

Testing Paragraph	Precision (%)	Recall (%)	F-score (%)
A	97.16	98.78	97.96
B	95.62	95.93	95.77
C	93.89	93.92	93.91

7. CONCLUSION

The preposition checker system for Myanmar language which can handle three types of preposition errors. Transformation Based Learning (TBL) Algorithm is applied for correction module and Decision Tree (DT) is used for rule generation module. TBL rules must follow patterns, called templates that are meant to capture the relevant feature combinations and DT learning has the ability to automatically select good feature combinations. The proposed algorithm is very useful in checking preposition errors of Myanmar language.

This system emphasized on Myanmar sentences which follow Myanmar grammar rules and it cannot handle Parli words. This system can be applied in Myanmar NLP applications. This system can be extended to correct conjunction and particle errors of Myanmar sentences which are ambiguous for poor readers and non-native learner. This system can be applied in Myanmar NLP applications. Evaluation results show that this system can provide promising accuracy.

8. REFERENCES

[1] A.M. Mon, Spell Checker for Myanmar Language, Proceeding of 4th Applied Information and Communication Technology Conference, 2011.

[2] A. D. Matthieu Hermet and S. Szpakowicz, \Using the web as a linguistic resource to automatically correct lexico-syntactic errors," in LREC'08, (Marrakech, Morocco), May 2008.

[3] Carberry, S, Vijay-Shanker, K., Wilson, A., and Samuel, K. (2001) Randomized rule selection in transformation-based learning: a comparative study. Natural Language Engineering, 7(2):99-116.

[4] Corston-Oliver, S., Gamon, M.: Combining decision trees and transformation-based learning to correct transferred linguistic representations. In: Proceedings of the Ninth Machine Translation Summit, pp. 55–62. Association for Machine Translation in the Americas, New Orleans (2003)

[5] 4. Carberry, S., Vijay-Shanker, K., Wilson, A., Samuel, K.: Randomized rule selection in transformation-based learning: a comparative study. Nat. Lang. Eng. 7(2), 99–116 (2001). doi:10.1017/S1351324901002662

[6] J. Eeg-olofsson and O. Knutsson, \Automatic grammar checking for second language learners - the use of prepositions," in NoDaLiDa, (Reykjavik, Iceland), 2003.

[7] Lidia Mangu and Eric Brill, Automatic Rule Acquisition for Spelling Correction

[8] Milidiú, R.L., Duarte, J.C., dos Santos, C.N.: TBLtemplate selection: an evolutionary approach.

[9] In: Proceedings of Conference of the Spanish Association for Artificial Intelligence—CAEPIA, Salamanca (2007)

[10] Myanmar Words Commonly Misspelled and Misused Book, Department of Myanmar Language commission, Ministry of education, Union of Myanmar July, 2003.

[11] Myanmar Grammar, Department of Myanmar Language commission, Ministry of education, Union of Myanmar June 2005.

[12] Phyu Hninn Myint, Tin Myat Htwe and Ni Lar Thein, "Assigning Automatically Part-of-Speech Tags To Build Tagged Corpus for Myanmar Language", The Fifth Conference on Parallel Soft Computing (PSC 2010), Yangon, Myanmar, December 16, 2010

[13] T. Latter, "A Grammar of the language of Burma", Baptist Mission Press, 1845.

[14] T. S. KO, "Elementary Handbook of the Burmese language", Rangoon: American Baptist Mission Press, 1924.

[15] မြန်မာစာ မြန်မာစကား , Department of Myanmar Language commission, Ministry of education, Union of Myanmar June 2007

[16] နည်းသစ် မြန်မာသဒ္ဒါ , Department of Myanmar Language commission, Ministry of education, Union of Myanmar

[17] Brill, E.: Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging. Comput. Linguist. 21(4), 543–565 (1995)

[18] Florian, R., Henderson, J.C., Ngai, and G.: Coaxing confidences from an old friend: probabilistic classifications from transformation rule lists. In: Proceedings of Joint Sigdat Conference on Empirical Methods in NLP and Very Large Corpora. Hong Kong University of Science and Technology, Kowloon (2000)