Automatic Template Generation for Myanmar Preposition Checking System

Khaing Htet Win
University of Computer Studies, Yangon
Khainghtetwin86@gmail.com

Abstract

Although the numbers of researches are increased in natural language processing, the development of grammar checking system for Myanmar language is very few. Much of the researches in checking area depends on hand-crafted rules and focuses on certain error types. However, the manual process of creation of rules is costly, time-consuming and error-prone. It seems therefore advisable to use machine-learning algorithms to create the rules automatically. In this paper, we describe a system for correcting Myanmar preposition errors automatically and non-native writers often make these errors. A decision tree algorithm (DT) is applied for automatic template generation of preposition errors using Myanmar Text Corpus. Correct Sentences in the corpus increase the performance of the method. The method works automatically, does not require problem domain expert to build these rule and can be applied to Myanmar language text.

***Keywords**: Myanmar Prepositions Checker, Templates,Decision Tree (DT)*

1. Introduction

In linguistic, preposition is a set of logical and structural rule that are important because they work to connect various parts of a sentence.Prepositions tell where something is in relation to something else or how things arerelated inspace(ထဲတွင်၊ ပေါ်တွင်၊ သို့၊ နှင့်) (in, on, to, with). Others show physical connections or possession between nouns or pronouns and another word such as(၏၊ နှင့်၊ ဖြင့်) (of, with, by).

Prepositions are known to be one of the most frequent sources of errors for Myanmar language. As a result, it seems desirable to focus on this problematic part of language, in developing system for automatic error correction in Myanmar language writhing. 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. For example, "ကျောင်းအုပ်ကြီး မှ ဆုချီးမြှင့်သည်"and "ကျောင်းအုပ်ကြီးကဆုချီးမြှင့်သည်။" are seemed grammatically correct.

Preposition errors can generally result from the mistake made by human when he/she confuses with many usages. Generally, these errors can be distinguished into three groups :(1) Missing Preposition Error(2)Misused Preposition Error(3) Unwanted Preposition Error. Prepositions are so difficult to master because they perform so many complex roles. In Myanmar language, prepositions appear in adjuncts, they mark the arguments of predicates, and they combine with other parts of speech to express new meanings. In the preposition error correction methods, Decision tree algorithm and Transformation Based Learning algorithm are proposed.

This paper is organized as follow: Sections 2 presents brief over of the related works. Our proposed method described in Section3. Evaluation and experimental results are discussed in Section 4. We conclude in Section5.

2. Related Works

It is one of the difficult problems for learners to use prepositions properly. Izumi et at. [4]reported error rates for English prepositions that were as high as 10 % in a Japanese learner's corpus. Felice et al. [9] also reported that 12% of errors were prepositions in a small error-tagged corpus they created.

Izumi et al. [5] used a maximum entropy approach to recognize various errors using contextual features. They do not show performance of prepositions specifically, but overall performance of the targeted 13 error types achieved 25% precision and 7% recall.

Gamonet al. [1] proposed a complex system including a languagemodel and decision trees to detect preposition and determiner errors. Their system performed at 79% precision, butrecall was not shown. Tetreault et al. [8] used a maximumentropy classifier to build a model of correct preposition usage for 34 common English prepositions. They reported 84% precision and 19% recall. And, Felice et al. [9] used a maximum entropy approach to correct preposition and

determiner errors. They reported 70% accuracy of preposition classifying in native English writing.

A grammar correction system for Danish has been implemented by [3]. It corrects two problems in the Danish language: article-noun agreement and comma placement. This is in fact a limited monolingual language improver. Errors are generated in a semi-random way in an existing corpus, and TBL constructs rules to fix these errors. With a parallel corpus and a translation, the translation can be improved in the same way.

An evolutionary approach based on Genetic Algorithm (GA) to automatically generate TBL templates is presented in [6]. 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 this paper, we propose a system for correcting preposition errors automatically using Decision Tree (DT) Learning. The system is mainly based on the automatic template correction algorithm. Templates are extracted fromcorpora of error free text.

3. Types of Preposition Errors

Noun preposition error can result generally from the mistake made by human when he/she confuse in many usage. Generally, human-generated errors can be distinguished into three groups.

(1) Missing Preposition

➔

(He beat the cat with stick.)

(2) Misused Preposition

➔

(Teacher gives a book to Mg Mg.)

(3) Unwanted Preposition

➔

(Mango is the best fruit among fruits.)

4. Decision Tree Algorithm

Decision tree learning is one of the most widely used and practical methods for inductive inference. It is a method for approximating discrete-valued functions that is robust to noisy data and capable of learning disjunctive expressions.

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Learned trees can also be re-represented as sets of if-then rules to improve human readability. These learning methods are among of most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks.

In this system, decision tree is applied for template generation module. Checking module 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 system is very useful in checking preposition errors of Myanmar language.

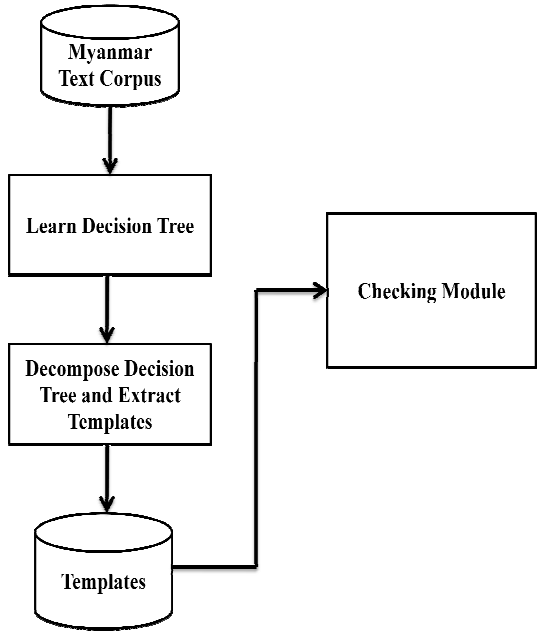


Figure 1. Flow of Automatic Template Generation Module

Figure 1 shows the process flow of automatic Template Generation Module. When this process gets correct Myanmar sentences as input, it first parses the sentence with Parser [7]and tag with basic POS tagging category. Using the Decision Tree Algorithm, the process built tree for template generation from a large corpus. And then, the tree is decomposed as depth-first traversal. For each tree node, template is extracted in the path from root to this node. Finally, templates are stored in the database and these templates are input into Rule Derivation of Checking Module. We describe details of automatic template generation process below.

4.1 Myanmar Text Corpus

Myanmar Text Corpus is built manually. We extended the functional annotated tagged corpus that is proposed in [7]. We added sentences from newspapers and historical books of Myanmar to the existing corpus. The corpus consists of approximately 5000 sentences with average word length 1 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. Myanmar textbooks and historical books are text collections, as shown in Table 1. In our corpus, a sentence contains Myanmar word and its POS tag with category. Figure 2 shows the example corpus sentence.

Table1. Corpus Statistics

Text Type	Sentences
Myanmar Grammar book	1450
Myanmar Text book of middle and high school	1900
Myanmar newspapers	1150
Others	500
Total	5000

PRN.Person# NN.Building# PPM.Destination # VB.Common # SF.Declarative

NN.Person# Part.Number# PPM.Extract# PRN.Person# Part.Common# JJ.Dem# Part.Common# VB.Common# SF.Declarative

PRN.Distobj # NN.Common # PPM.Obj # PRN.Question # VB.Common # art.Support # SF.Interrogative

Figure 2.Example of Myanmar Text Corpus

4.2Template Generation Module

The correct POS tagged sentences are used to construct the decision tree. As an example,input sentences are as follow:

1. PRN.Person# NN.Building# PPM.Destination # VB.Common # SF.Declarative
2. NN.Person# Part.Number# PPM.Extract# PRN.Person# Part.Common# JJ.Dem# Part.Common# VB.Common# SF.Declarative

The DT performs a partitioning of this corpus using the following equation

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

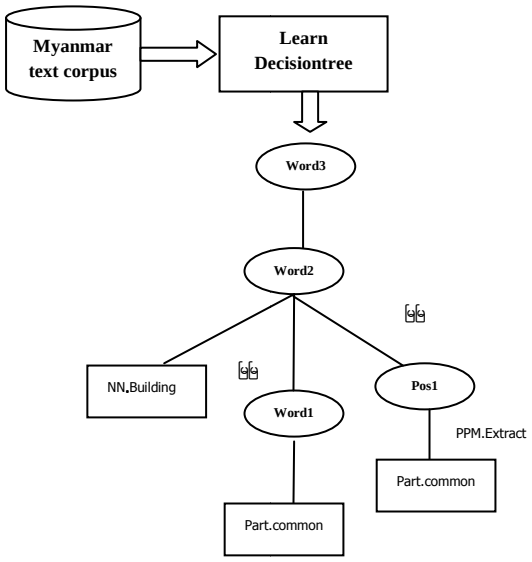


Figure 3. Decision Tree Learning

Where, C_i is a class label from C,|C| is the number of classes andPT (C_i) is estimated by the percentage of examples belonging to c_i in Corpus.

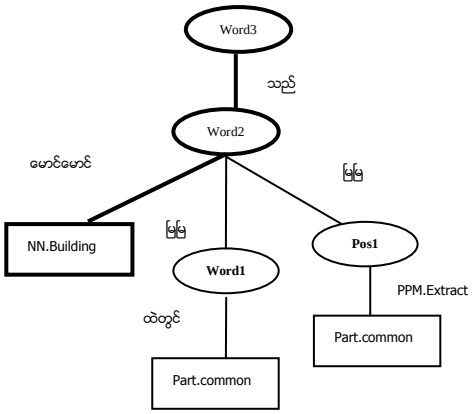
And then, information gain IG (T,A) can be used in order to select the root node of decision tree.

$$IG(T, A) = H(T) - \sum_{v \in Value(A)} \frac{|T_v|}{|T|} H(T_v)$$

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.

In Figure 3, the DT induction process for Myanmar Preposition Checking is illustrated. The five selected features are: word[3], the lexical item of next third word; word[2], the lexical item of next word; word[1], the lexical item of current word; pos[1], the POS tag of next word and the final block will become pos[4]. The feature values are shown in the figure as arc labels. The pruned trees are used in all experiments

that are shown here:



- **Word3= သည် word2= မောင်မောင် pos4= NN.Building**
- Word3= သည် word2= မြမြ word1= ထဲတွင် pos4= Part.Common
- Word3= သည် word2= မြမြ pos1= PPM.Extract pos4= Part.Common

Figure 4. Decision Template Extraction

In a DT, the more informative features appear closer to the root. Since the most promising templates are wanted to generate, the more informative features are first combined. Hence, the DT are traversed from the root to a leaf and features are collected in this path. This feature combination provides an information gain driven templates. Additionally, paths from the root to internal nodes also provide good templates. Figure 4 illustrates the template extraction process. In this figure, the template in bold is extracted from the tree path in bold.

4.3 Checking Module

Transformation Based error-driven Learning (TBL) is a successful machine learning algorithm. It has since been used for several Natural Language Processing tasks, such as part-of-speech (POS) tagging, English text chunking, spelling correction and noun-phrase chunking.

TBL algorithm is used in checking module. The requirements of the algorithm are:

- Myanmar Text Corpus, one that has been correctly tagged with POS tag set and another that remains unlabeled;
- An initial classifier, the baseline system, which tags the Myanmar sentence with basic POS tag sets.
- A set of rule templates, which are meant to capture the relevant feature combinations that would determine the sample's classification. Concrete 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:

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. instantiateRule (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 CandidateRules do
15. count Corrections (rule, POS Tagged Sentences) → good
16. count Errors (rule, POS Tagged Sentences) → 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

5. Experiment Results

This paper emphasizes on the preposition checking that can make the most error percentages of Myanmar Sentences. The system was first trained on a corpus containing 5000 Myanmar sentences collected from Myanmar Grammar Text Book. Testing of the system carried out on a different sentences containing preposition error collected over the non-native learners and students. The testing sentences are 750 sentences with different and possible preposition errors.

Three testing paragraphs are used for evaluation in ordertocalculate the accuracy of the preposition checker and each paragraph contains 250 sentences. First 16% preposition errors of the total words in the Second paragraph B has 39% preposition 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 total number of sentence. F-score is the mean of recall and precision, that is F = 2PR/ (P+R). The following figures correctly detected on the testing sentences with Myanmar Preposition Checker. In these figures of Average accuracy of overall system gets 95% precision, 92.33% recall and 93% f-score.

Table2. Experimental Results for Missing Error

Testing Paragraph	Precision (%)	Recall (%)	F-Score(%)
A	90.16	91.78	90.96
B	89.62	88.93	89.77
C	84.89	83.92	83.91

Table3. Experimental Results for Misused Error

Testing Paragraph	Precision (%)	Recall (%)	F-Score(%)
A	92.16	90.78	91.96
B	89.62	88.93	89.77
C	85.98	83.92	85.91

Table4. Experimental Results for Unwanted Error

Testing Paragraph	Precision (%)	Recall (%)	F-Score(%)
A	95.16	94.78	95.96
B	88.62	89.93	85.77
C	83.89	82.92	83.91

6. Conclusion

A preposition checker for Myanmar language which can handle three types of preposition errors. Decision Tree (DT)Algorithm is used forautomatic template generation module.DT learning has the ability to automatically select good feature combinations. For resources, a Myanmar Text Corpus, a Myanmar basic POS tagged corpus and “Myanmar Grammar” books published by Myanmar Language Commission. This system emphasized on Myanmar sentences which follow Myanmar grammar rules and it cannot handle Parli-words.

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.

References

[1] D. MatthieuHermet and S. Szpakowicz, “Using the web as a linguistic resource to automatically correct lexicon syntactic errors,” in LREC’08,(Marrakech, Morocco), May 2008.

[2] 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.

[3] D. Hardt, “Transformation-based learning of Danish grammar correction”, Proceedings ofRANLP, 2001

[4] E. Izumia, K. Uchimotoa, and H. Isaharaa, “SSTspeech corpus of Japanese learners’ English and automatic detection of learners’ errors,” ICAMEJournal, vol. 28, pp. 31–48, 2004.

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

[6] Milidiu`, R. L., Duarte, J. C., and dos Sandos, C. N. (2007). Tbl template selection: An evolutionary approach. In Proceedings of Conference of the Spanish Association for Artificial Intelligence-CAEPIA, Salamanca, Spain.

[7] PhyuHninMyint, Bigram Part-of-Speech Tagger for Myanmar Language

[8] S. Bergsma, D. Lin, and R. Goebel, “Web-scalen-gram models for lexical disambiguation,” in

IJCAI'09, (Pasadena, California), pp. 1507–1512, July 2009

[9] T. Brants and A. Franz, "Web 1T 5-gram corpus version 1.1," tech. rep., Google Research, 2006.

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

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