Transformation Based Learning Algorithm in Myanmar Preposition Checker

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

Abstract

Preposition checking and correction comprise of the primary problems in the area of Natural language Processing (NLP). Because they can appear to have an idiosyncratic behavior and which does not follow any predictable pattern even across nearly identical contexts. Transformation Based Learning (TBL) is a Machine learning technique frequently used in some Natural Language Processing (NLP) tasks. By adapting transformation based learning to the problem of preposition checking, new rules are produced from templates. TBL uses rule templates to identify error-correcting patterns and it generates a set of transformation rules that correct errors of an input text. The errors of preposition in the sentences are reduced with transformation-based learning. In this paper, a system is implemented for correcting Myanmar preposition errors and non-native writers often make these errors.

Keywords: Natural Language Processing (NLP), Preposition Checking System, Transformation Based Learning (TBL), Khaing Htet Win.

1. Introduction

Checking system is one of the most widely used tools within natural language engineering applications. Most of the word processing systems available in the market incorporate spelling, grammar, and style-checking systems for English and other widely used languages. Although, spell checking has been addressed for most of the Myanmar languages, still preposition checking systems are lacking.

Three main approaches are widely used for checking in a language; syntax-based checking, statistics-based checking and rule-based checking. In syntax based checking, each sentence is completely parsed to check the grammatical correctness of it. The text is considered incorrect it the syntactic parsing fails. In statistics-based approach, POS tag sequences are built from an annotated corpus, and the frequency, and thus the probability, of these sequences are noted. The text is considered incorrect if the POS-tagged text contains POS sequences with frequencies. The statistics based approach essentially learns the rules from the tagged training corpus. In rule-based approach, the approach is very similar to the statistics based one, except that the rules must be handcrafted [5].

Our approach falls in this third category and it uses transformation based learning (TBL) algorithm. The rules of TBL are more readily interpretable and can detect and suggest rectifications for a number of preposition errors in Myanmar sentences.

The rest of this paper is organized as follows: Section 2 describes the related works. In section 3, we describe the corpus creation. Sections 4 and 5 discuss the algorithms of transformation based learning and system architecture, respectively. Section 6 describes the experiment results. Finally, Section 7 shows the limitations and conclusion of Preposition Checking System.

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.

Gamon et al. [1] proposed a complex system including a language model and decision trees to detect preposition and determiner errors. Their system performed at 79% precision, but recall was not shown. Tetreault et al. [8] used a maximum entropy 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 efficiency 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 give an account of a system for correcting preposition errors automatically using Transformation based Learning. The system is mainly based on the automatic rule correction algorithm using templates. Templates are extracted using entropy calculation from the corpus.

3. Corpus Creation

Myanmar Text Corpus is to be built manually. The corpus consists of approximately 5000 sentences with average word length 12 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 grammar books are text collections, as shown in Table 1.

Table 1: Corpus Statistics

Text Type	Sentences
Myanmar Grammar book	1450
Myanmar Text book of school	1900

Myanmar newspapers	1150
Others	500
Total	5000

4. Transformation Based Learning

Transformation Based error-driven Learning (TBL) uses a greedy error correcting strategy. It has been used to learn rules for many NLP tasks, such as POS Tagging, Prepositional Phrase Attachment, Text Chunking, Spelling Correction and Dialogue Act Tagging. Its main objective is to generate an ordered list of rules that correct classification mistakes in the training set, which have been produced by a baseline system. The requirements of the TBL algorithm are: (1) Training corpus (2) An initial classifier and (3) A set of rule templates.

4.1 Transformation Based Learning Algorithm

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. Starts applying the baseline system, in order to guess an initial classification for the unlabeled version of the training set;
2. Compares the resulting classification with the correct one and whenever a classification 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. Compute the rules' scores (error repaired-error created). If there is not a rule with a score above an arbitrary threshold, the learning process is stopped;
4. Selects the best scoring rule, stores it in the set of learned rules and applies it to the training set;
5. Return to step 2.

5. System Architecture

The design of this preposition checking system is provided below in figure 1. A sketchy idea of this proposed design is provided below in terms of how the input text is processed to find potential preposition errors.

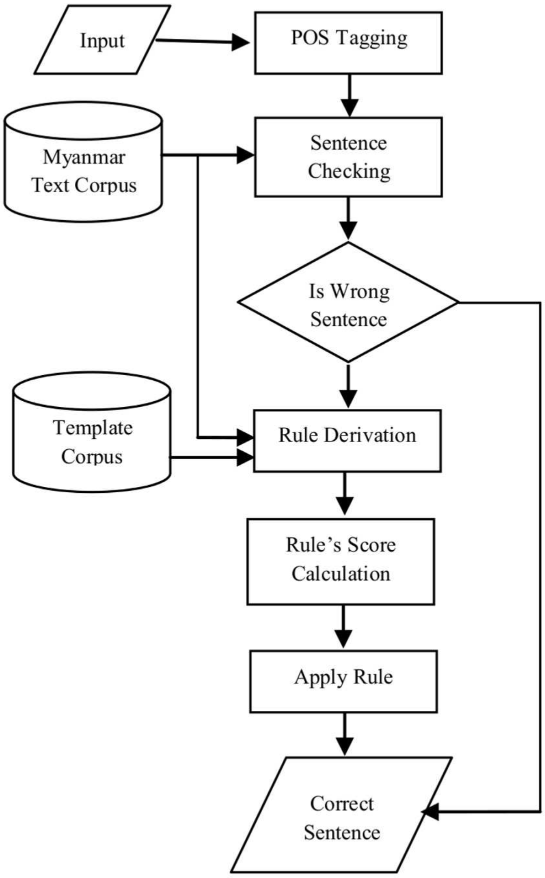

Figure 1: System architecture of preposition checking system

For preposition checking, the input text (Myanmar Sentence) is first given to a POS

Tagging process, which gives the appropriate pos tags for each word in sentence. For example:

Input:

သူတို့ ကျောင်းသွားကြသည်

(They go the school.)

Pos Tag Sentence:

သူ@PRN.Person#တို@Part.Number#ကျောင်း@NN.Building#သွား@VB.Common#ကြ@Part.Common#သည်@SF.Declarative

Then example sentence with the POS tags moves on to a Sentence Checking process, which attempts to compare the sentence using Myanmar Text Corpus.

Example of Myanmar Text Corpus:

သူ@PRN.Person#တို@Part.Number#ကျောင်း@NN.Building#သို@PPM.Direction#သွား@VB.Common#ကြ@Part.Common#သည်@SF.Declarative

ကျွန်မ@PRN.Person#တို@Part.Number#ရန်ကုန်@NN.Location#သို@PPM.Direction#သွား@VB.Common#ကြ@Part.Common#သည်@SF.Declarative

The checking process messaged that "It is wrong sentence!". Because the input sentence is not exit in the corpus.

Therefore, this sentence is passed on to a Rule Derivation, which builds rules using templates that are produced from decision tree. The template generation is done by entropy calculation. [10]. These following rules are generated from process of Rule Derivation.

Rule1 : Add သို့ @PPM.Direction between [-1] and [0] if pos[1]= Part.Common and pos[-1] = NN.Building

Rule2 : Add သို့ @PPM.Direction between [-1] and [0] if pos[2] = SF.Declarative and pos[0] = VB.Common

Rule3: Add သို့ @PPM.Direction between [-1] and [0] if pos [-1] = NN.Building

The error detection rules are to detect potential errors in the text and provide correction to resolve that errors. It displays an error message if some grammatical information fails to match. To resolve this error, the error checking module will use transformation based learning algorithm to generate the correct form. The central idea in the TBL learning process is to greedily learn rules that incrementally reduce the number of errors from the input sentence. At each iteration, the algorithm learns the rule that has the highest score. The score of a rule r is the difference between the number of errors that r repairs and the number of errors that are creates. So the rules' scores are computed (Good - Bad) as follows:

Rule 1: Good = 1 Bad = 0; Score= 1

Rule 2: Good = 5 Bad= 1; Score=4 (Best Rule)

Rule 3: Good = 1 Bad=1; Score = 0

If there is not a rule with a score above an arbitrary threshold, the learning process is stopped.

Apply Rule:

သူ@PRN.Person#တို@Part.Number#ကျောင်း@NN.Building#သို@PPM.Direction#သွား@VB.Common#ကြ@Part.Common#သည်@SF.Declarative

Correct Sentence:

သူတို့ ကျောင်း သို့ သွားကြသည်

(They go to the school)

Finally, Selects the best scoring rule, and applies it to the input sentence and produce correct sentences.

6. Experiment Results

This paper emphasizes on the preposition checking that can make 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 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

7. Conclusion

A preposition checker of Myanmar language which can handle three types of errors: (1) Missing Preposition Error (2) Misused Preposition Error (3) Unwanted Preposition Error. For resources, a Myanmar Text 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] A. D. Matthieu Hermet 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 of RANLP, 2001

[4] E. Izumia, K. Uchimotoa, and H. Isaharaa, "SST speech 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] Phyu Hnin Myint, 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] Khaing Htet Win, Automatic Template Generation for Myanmar Preposition Checking System Proceedings of the 12th International Conference on Computer Applications (ICCA 2013), Yangon, Myanmar, 2013.