Developing a Hybrid Approach for English Grammar Checker

Nay Yee Lin
University of Computer Studies, Yangon
nayyeelynn@gmail.com

Abstract

English becomes an essential language for communication all over the world. As English is a universal language, most of the researchers are concentrated on the development of English language. We propose an approach that develops a grammar checker for English as a second language (ESL) and introduce statistical and rule based approach to detect grammatical errors in sentences. In order to complete English grammar checking, we need to take three main functions such as detecting the sentence rule, analyzing the chunks errors and correcting the grammatical sentence. In this paper, we have presented a hybrid approach that uses context free grammar based bottom up parsing, trigram based markov model and rule based model. The proposed system is concerned with the target language generation to solve distortion, deficiency and make smooth the translated English sentences.

1. Introduction

Machine Translation Systems expect target language output to be grammatically correct within the frame of proper grammatical category. In Myanmar-English machine translation system, source language model, alignment model, translation model and target language model are required to complete translation. All of these models, our proposed system is concerned with the target language generation to solve distortion, deficiency and make smooth the translated English sentences. In order to smooth the Myanmar-to-English translation, we propose a hybrid approach for developing an English grammar checker.

Grammar checking is one of the most widely used tools within natural language processing applications. Grammar checker determines the syntactical correctness of a sentence. Grammar checking is mostly used in word processors and compilers. Although all major Open Source word processors offer spell checking and grammar checker feature. Such a feature is not available as a separate free program either for machine translation. Therefore, our approach is a free program which can be used both as a stand-alone grammar checker.

Three methods are widely used for grammar checking in a language; syntax-based checking, statistics-based checking and rule-based checking. In syntax based grammar checking, each sentence is completely parsed to check the grammatical correctness of it. The text is considered incorrect if 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 lower than some threshold. 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 [12].

There are a variety of techniques for Grammar checking. Among them, we build a grammar checker by using hybrid approach which considers the context free grammar (CFG) based bottom up parser to decide whether the sentence rule is correct or not, the trigram markov model to analyze the chunk errors and the rule-based model to complete the translated sentence grammatically correctness.

This paper is organized as follows. Section 2 describes the related work. In section 3, the proposed system is described. Section 4 discusses the problem description. Finally section 5 concludes the paper.

2. Related Work

The detection, analyzing and correction of errors for English language are vital keys for the development of NLP applications such as Machine Translation. Several researchers worked the grammar checking in natural language processing for various languages.

G. Adriaens [7] described the ongoing developments in the LRE-2 project SECC (A Simplified English Grammar and Style Checker/Corrector). SECC's main goal is the development of a tool for technical writers who produce documents in a variant of Simplified English (SE). The tool will check if the documents comply with the syntactic and lexical rules; if not, error messages are given, and automatic correction is attempted wherever possible to reduce the amount of human correction needed.

Kathleen F. McCoy, Christopher A. Pennington and Linda Z. Suri [6] considered a user model which can be tailored to different types of users in order to identify and correct English language errors. It is

presented in the context of a written English tutoring system for deaf people who use American Sign Language. The model consists of a static model of the expected language (represented on a feature by feature basis) and a dynamic model that represents how a language might be acquired over time. Together these models affect scores on a set of grammar rules which are used to produce a "best interpretation" of the user's input.

Mandeep Singh Gill, Gurpreet Singh Lehal and Shiv Sharma Joshi [11] provided the grammar checking software developed for detecting the grammatical errors in Punjabi texts and providing suggestions wherever appropriate to rectify those errors. This system utilizes a full-form lexicon for morphology analysis and rule-based systems for part of speech tagging and phrase chunking. The system supported by a set of carefully devised error detection rules can detect and suggest rectifications for a number of grammatical errors, resulting from lack of agreement, order of words in various phrases etc., in literary style Punjabi texts.

Anuradha Sharma, Nishtha Jaiswal [1] handled a problem is to convert ill framed sentence to nearest appropriate grammatical structures. They have used a major corpus in tourism and health domains (about 14 lakhs words collected from the web). They formed structures of English practiced mostly in India have been identified to design the predictor. This was incorporated in the AnglaBharti Engine and gave significant improvement in the Machine Translation output.

B.M. Sagar, G Shobha and P. Kumar Ramakanth [2] developed a way of producing context free grammar for solving Noun and Verb agreement in Kannada Sentences. In most of the Indian languages including Kannada a verb ends with a token which indicates the gender of the person (Noun/ Pronoun). They showed the implementation of this agreement using Context Free Grammar. It used Recursive Descent Parser to parse the CFG. Around 200 sample sentences have taken to test the agreement.

3. Proposed System

This section describes the proposed system to offer a good grammar checker as shown in figure (1). In our proposed system, we need to take an English sentence as an input. Initially, input sentence can be tokenized and tagged POS to each word. POS tagging is the process of assigning a part-of-speech tag such as noun, verb, pronoun, preposition, adverb, adjective or other tags to each word in a sentence. POS-tagging is one of the main tools needed to develop any language corpus. In this system, Tree Tagger is used for POS tagging.

Secondly, groups these tagged words into chunks by using hand written chunk rules. Parsing chunks is an approach to natural language based on the understanding that the majority of sentences in the English language can be matched to the chunk rules. Chunking or shallow parsing segments a sentence into a sequence of syntactic constituents or chunks, i.e. sequences of adjacent words grouped on the basis of linguistic properties (Abney, 1996).

After that, our system checks these chunks relationship for input sentence by using sentence corpus. If the sentence rule is incorrect, we analyze the chunk patterns using hybrid approach with trigram model and rule based model. If the sentence rule is correct, then we check the grammar of each word and correct the sentence by English grammar rules.

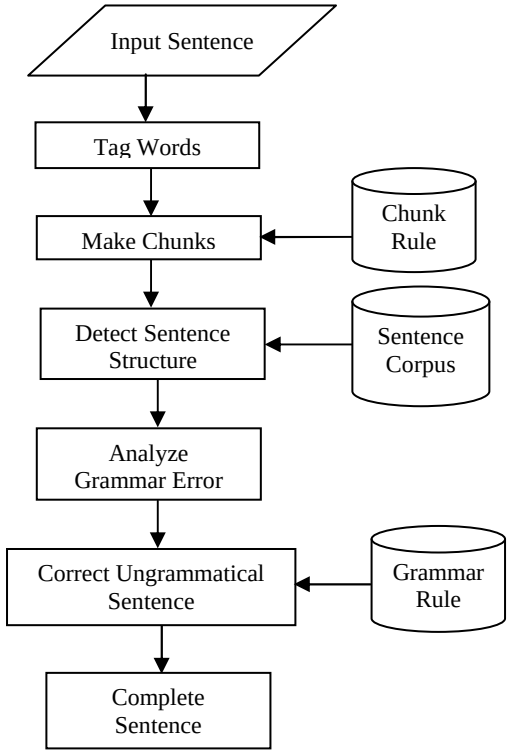

Figure (1) Overview of the proposed system

3.1. Detection Sentence Structure by using Context Free Grammar

Our proposed system use considerable knowledge about the syntax of language. Syntax means representation of sentence structure of a language. We identify the chunk types and detect the English sentence in chunk structure by using Context-free grammar (CFG).

CFG constitute an important class of grammars, with a broad range of applications including programming languages, natural language processing, bioinformatics and others. GFG's rules present a single symbol on the left-hand-side, are a sufficiently powerful formalism to describe most of the structure in natural language, while at the same

time is sufficiently restricted as to allow efficient parsing. Context Free Grammars are backbone of many models of the syntax of natural language [15].

A context-free grammar G = (V, Σ S, P) is given by

- A finite set V of variables or non terminal symbols.
- A finite set Σ of symbols or terminal symbols. We assume that the sets V and Σ are disjoint.
- A start symbol S $\in$ V.
- A finite set P $\subseteq$ V $\times$ (V $\cup$ T)* of productions. A production (A,α), where A$\in$ V and α$\in$ (V $\cup$ T)* is a sequence of terminals and variables, is written as A→α.

Context Free Grammars are powerful enough to express sophisticated relations among the words in a sentence. It is also tractable enough to be computed using parsing algorithms. Parsing is used to understand the syntax and semantics of a natural language sentences confined to the grammar.

There are two methods for parsing such as Top-down parsing and Bottom-up parsing. In Top–down parsing, begin with the start symbol and attempt to derive the input sentence by substituting the right hand side of productions for non terminals. In Bottom–up (shift–reduce) parsing, begin with the input sentence and attempt to work back to the start symbol. Bottom-up parsers handle a large class of grammars. In this system, Bottom-up parsing is used to detect the sentence grammar rule [4].

Bottom-up parser proceeds by assembling words into POS tagging, and making into higher level chunks, until a complete sentence has been found. A simple example is shown as follows:

The dog is under the tree.
<DT> dog is under the tree.
<DT><NN> is under the tree.
<DT><NN><VBZ>under the tree.
<DT><NN><VBZ><IN>the tree.
<DT><NN><VBZ><IN><DT> tree.
<DT><NN><VBZ><IN><DT> <NN>.
<DT><NN><VBZ><IN><DT> <NN><SENT>
NC_<VBZ><IN><DT> <NN><SENT>
NC_VC_<IN><DT> <NN><SENT>
NC_VC_PPC_<DT> <NN><SENT>
NC_VC_PPC_NC_<SENT>
NC_VC_PPC_NC_End

3.2. Analyzing Error by using Hybrid Model

The second function of the proposed system is analyzing chunk errors by using hybrid model (Trigram Markov Model and Rule Based Model). The purpose of the hybrid language model is to assign high probabilities to likely word sequences and to correct the sentence structure.

The simplest models of natural language are n gram Markov models. The Markov models for bigrams and trigrams (or for any n grams) are called Markov Chains. A Markov Chain is a Markov model for which there is at most one path through the model for any given input. In these models, the probability of each word depends on the n-1 words that precede it. The transition probabilities in n-gram models are estimated from the counts of word combinations in the training corpus [9].

N-gram and Trigram models are the examples of statistical model. In N-gram language model, each word depends probabilistically on the n-1 preceding words. This is expressed as shown in equation (1).

$$p(W_0, n) = \prod_{i=0}^{n-1} p(W_i | W_{i-n+1}, ..., W_{i-1}) \quad (1)$$

. When N is big, memory and processing power requirement is high. Good results are obtained by N=3. This is called trigram language model, where each word depends probabilistically on previous two words and is shown in equation (2) [9].

$$p(W_0, n) = \prod_{i=0}^{n-1} P(W_i | W_{i-1}, W_{i-2}) \quad (2)$$

Trigram language model is most suitable due to the capacity, coverage and computational power [3]. For shaping the trigram model into a greater level of suitability some advanced and optimizing techniques like smoothing, caching, skipping, clustering, sentence mixing, structuring and text normalization can be applied. This model makes use of the history events in assigning the current event some probability value and that suits our approach philosophy. We have favored Trigram Markov Model over other statistical models to analyse the required chunk types and rule based model to correct sentence rule.

For example, when a sample sentence *"A man a woman came to our house* "has been chunked, sentence rule may be NC_NC_VC_TO_NC_End. This is incorrect sentence structure by using sentence rules. Therefore our proposed system searches needed chunk by using statistical model.

The first NC of the input sentence is found in the sentence corpus and the probability is P(NC/none none) =0.58490566. The second NC does not found and the probability is P(NC/none NC)=0.0. We find the correct chunk in the second place by using the probabilities as follows:

P(VC/none NC)=0.547169811
P(RC/none NC)=0.018867925
P(COC/none NC)= 0.018867925

By these probabilities, RC, VC and COC can be in the second place. We substitute the second place with VC firstly as the maximum probability. Then we get the sentence rule as NC_VC_NC_VC_ TO_NC_End.

However, this rule is incorrect by using sentence rules. So we have to substitute RC and COC in the

second place. When COC is placed in the second place, we get the correct sentence rule NC_COC_NC_VC_TO_NC_End by comparing the sentence rules. Therefore, in this example, our system can search the correct chunk type by using trigram and rule based model.

3.3. Correcting the sentence by using Rule Based Model

Most of the machine translation software on the market today is rule-based. Rule-based approach has successfully used to develop natural language processing tools and applications. The accuracy of translation system can be increased by the product of the rule based correcting the ungrammatical sentences. In the rule-based approach, English grammatical rules are developed to define precisely how and where to assign the various words in a sentence. The final step of our proposed system is controlled by the grammar rules.

This function is used to query grammatical errors in translated English sentence. Rule-based system is more transparent: errors are easier to diagnose and debug. A rule based machine translation system consists of collection of rules called grammar rules, lexicon and software programs to process the rules. It is extensible and maintainable.

Rule based approach is the first strategy ever developed in the field of machine translation. Rules are written with linguistic knowledge gathered from linguists. Rules play major role in various stages of translation: syntactic processing, semantic interpretation, and contextual processing of language [14].

Rule-based approach relies on hand-constructed rules that are to be acquired from language specialists, requires only small amount of training data and development could be very time consuming. It can be used with both well-formed and ill-formed input. Grammatical rules describe sentence and phrase structures, and ensure the agreement relations between various elements in the sentence. There are different types of rules in this system. These rules can determine the chunk structures and syntactic structure and ensure the agreement relations between various chunks in the sentence [5].

The purpose of the rules is either to assign words depending on rules or, in the more common grammar approach, to remove illegitimate words in the sentence based on rules. The construction of a grammatical rule-based process can be a time-consuming task, since the number of rules is large.

4. Problem Description

This system proposes a target-dominant grammar checking for Myanmar-English machine translation system. Sentence analysis has been a critical problem in machine translation because of high complexity. The syntactic structure of a sentence is a necessary to determine its structure. Such structures assign a syntactic category (verb, noun, etc) to each word in the sentence and specify how these categories are clustered to form higher level categories (NC, VC etc) until building the whole sentence. The grammar specifies the permitted structures in a language.

In this system, there are nine main chunk types to parse the sentence as chunk levels as shown in Table (1).

Table (1) Chunk Types

NC	Noun Chunk
VC	Verb Chunk
AC	Adjective Chunk
RC	Adverb Chunk
PTC	Particle Chunk
PPC	Prepositional Chunk
COC	Conjunction Chunk
QC	Question Chunk
INFC	Infinitive Chunk

There are various sentence grammatical rules to correct the ungrammatical sentence. At present, this grammar checking system detects and provides corrections for the following grammatical errors, based on the study of English grammar related texts.

1) Chunk Omission
2) Subject Verb Agreement
3) Omitted Determiner
4) Inappropriate Determiner Formation
5) Inappropriate Noun use
6) Missing Markers (.,?)
7) Missing Capital
8) Incorrect Verb Form

5. Conclusion

In this paper, we have presented a hybrid approach that uses context free grammar based bottom up parsing, trigram based markov model and rule based model. Our proposed system describes detecting English language in chunk structure and sentence structure, providing the grammar errors for chunk level and correcting the ungrammatical input sentence. This system shows that hybrid statistical and rule-based approach is useful and can improve the effectiveness of automated grammar checking.

6. References

[1] Anuradha Sharma, Nishtha Jaiswal, "Reducing Errors in Translation using Pre-editor for Indian English Sentences", Proceedings of ASCNT–2010, CDAC, Noida, India, pp.70 – 76.

[2] B.M. Sagar, G. Shobha and P.Kumar Ramakanth, "Solving the Noun Phrase and Verb Phrase Agreement in Kannada Sentences", International Journal of Computer Theory and Engineering, Vol. 1, No. 3, August, 2009, 1793-8201.

[3] Brian Roark Eugene Charniak, "Measuring Efficiency in High-Accuracy, Broad-Coverage Statistical Parsing" Proceedings of the COLING 2000, Workshop on Efficiency in Large-Scale Parsing Systems, 2001, Pages 29-36.

[4] D.Cooper Keith, Ken Kennedy, Linda Torczon, "Bottom-up Parsing", 2003.

[5] Daniel Naber, "A Rule-Based Style and Grammar Checker",2003

[6] F.McCoy Kathleen, A.Pennington Christopher, Z.Suri Linda,"English Error Correction: A Syntactic User Model Based on Principled Mal-Rule Scoring".Computer and Information Sciences Department and Applied Science and Engineering Laboratories, University of Delaware.

[7] G. Adriaens, "Simplified English Grammar and Style Correction in an MT Framework", Translation and the Computer 15, Papers at a conference…18-19, November, 1993 (London:Aslib).

[8] Laurie Buscail, and Patrick Saint-Dizier, "Textual and Stylistic Error Detection and Correction: Categorization, Annotation and Correction Strategies", IEEE English International Symposium on Natural Language Processing, 2009.

[9] Lawrence Saul and Fernando Pereira, "Aggregate and mixed order Markov models for statistical language processing", AT&T Labs–Research,180 Park Ave, D-130,Florham Park, NJ 07932.

[10] M Selvam, A M Natarajan, and R Thangarajan, "Structural Parsing of Natural Language Text in Tamil Using Phrase Structure Hybrid Language Model", International Journal of Computer, Information and Systems Science, and Engineering 2:4 2008.

[11] Mandeep Singh Gill, Gurpreet Singh Lehal, Shiv Sharma Joshi, "A Punjabi Grammar Checker".

[12] Md. Jahangir Alam, Naushad UzZaman and Mumit Khan, "N-gram based Statistical Grammar Checker for Bangla and English", Center for Research on Bangla Language Processing, BRAC University, Dhaka, Bangladesh.

[13] P.Abney Steven, Parsing By Chunks, Bell Communication Research, November 10, 1994.

[14] Paisarn Charoenpornsawat, Virach Sornlertlamvanich, Thatsanee Charoenporn, "Improving Translation Quality of Rule-Based Machine Translation", Information Research and Development Division,National Electronics and Computer Technology Center, Thailand.

[15] Ramki Thurimella, "Context Free Grammars", 2005.