Myanmar Spell Checker

Aye Myat Mon[1], Thandar Thein[2]

[1, 2] University of Computer Studies, Mandalay, Myanmar
[1]*Amyatmon99@gmail.com*, [2]*thandartheinn@gmail.com*

Abstract: ***Natural Language Processing (NLP) is one of the most important research area carried out in the world of Human Language. For every language, spell checker is an essential component of many of Office Automation Systems and Machine Translation Systems. In this paper, we develop a Myanmar Spell Checker System which can handle Typographic errors, Sequence errors, Phonetic errors, and Context errors. A Myanmar Text Corpus is created for developing Myanmar Spell checker. To check Typographic Errors, corpus look up approach is applied. Myanmar3 Unicode is applied in this system so that it can automatically reorder the character sequence. A compound misused word detection algorithm is proposed for Phonetic Errors checking and Bayesian Classifier is applied for Context Errors checking. In this system, Levenshtein Distance Algorithm is applied to improve users' efficiency by providing a suggestion list for misspelled Myanmar Words. We provide evaluation results of the system and our approach can handle various types of Myanmar spell errors.***

Keywords: Levenshtein Distance Algorithm, Myanmar Spell Checker, Myanmar Text Corpus, Natural Language Processing, Naïve Bayesian Classifier

1. Introduction

Spell checking is one of the most vital and widely studied NLP tasks, which is used in orde r to increase the success rate of NLP applications. Many NLP applications like Machine Translation Systems, Text to Speech Systems and Information Retrieval Systems require automated spell checking of text. Many different techniques for detection and correction of spelling errors are based on English. Since every language has its own writing system, the techniques that perform well for one language, may not perform that well for some other languages and they may even totally fail. English spell checker will fail on the first step of recognizing Myanmar word boundaries because in Myanmar, unlike English, word boundaries are not marked with spaces. Human language translation is a difficult task for natural language because there has language ambiguity and varies according to their features and nature. Myanmar word transformations are similar to other Asian Language including Indian, Japanese, Thai and Chinese Language. In our country, Myanmar Language is used as an official language so spell checker is an essential role for the development of Machine Translation system.

Myanmar is also among the languages whose writing system is different from that of English and therefore existing techniques cannot be applied for Myanmar spell checking. Myanmar word does not have white space between words so it is difficult to tokenize. Although each Myanmar word can be identified by word boundary correctly, all words may not have meanings because they are not in the dictionary. The most common reasons for misspelled and misused words are caused by phonetic similarity and typing error of Myanmar characters. The categories of error words are: (i) *Typographic Error* which is mistyped the key in the wrong order and accidentally type characters (ii) *Phonetic Error* which is pronounced the same as the intended word but the spelling is wrong (iii) *Sequence Error* which often caused the wrong format of character sequence and (iv) *Context Error* which is pronounced the same as the intended word but the word is ambiguous for the input sentence.

In this work, first we study the details on Myanmar Language to identify the problem area of Myanmar spell errors and then we develop Myanmar spell checker. It consists of two phases: spell errors detection and suggestion list generation.

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 describes nature and collation of Myanmar Language. Section 4 depicts spell error patterns and Section 5 describes the proposed spell checker framework. Implementation of Myanmar Spell Checker System is presented in Section 6. Experimental results are depicted in Section 7. Finally, conclusion on this work is given in Section 8.

2. Related Work

Many researchers have been worked for spell checker of Asian Languages. Even though other Asian spell checker researches have been done for two decades, Myanmar spell checker research is still in its infancy. There is a very little amount of work done in this field. In this section, we discuss briefly some of the related work and history in the area of spell checking and suggestion generation.

Adbullah et.al [13] proposed an alternative approach to check the spelling of Bangla text that used Finite State Automata (FSA) to probabilistically crate the suggestion list for a misspelled word. They used backtracking to add each possible solution to the suggestion list. Their system was only handled non-word errors in Bangla text. UzZaman et.al [11] proposed for generating suggestion for typographical errors with the edit distance of 2 from the misspelled word, which obviates the need for computing the edit distances of the entire lexicon from the misspelled word. In [8], their phonetic encoding was based on the Soundex algorithm, modified to match Bangla phonetics. Their approaches used by PHONIX and Metaphone variants do provide some contexts. Their encoding is equally applicable in a wide range of text processing applications, from searching for patient records in medical database to matching names in census record. Dhanabalan et.al [6] presented Tamil Spell Checker by providing possible suggestions for erroneous

words. User has the provision to select the suggestion among the list, ignore the suggestion or add the particular word to the dictionary. Htay et.al [15] presented Myanmar word segmentation using syllable level longest matching approach. They used a combination of stored lists, suffix removal, morphological analysis and syllable level n-grams to hypothesize valid words with about 99% accuracy. The author [2] presented an approach consists of an approximate word matching method, an N-best word segmentation algorithm and used a statistical language model. Word-based correction method is proposed for Optical Character Recognition errors. It outperforms the conventional character based correction method. Fassati et.al [14] addressed the problem of real word spell checking and proposed a methodology based on a mixed trigrams language model. Their model had been trained and tested with data from the Penn Treebank. Their approach has been evaluated in terms of hit rate, false positive rate and coverage.

Golding [3] proposed a hybrid method for context sensitive spelling correction by combining Bayesian classifier and decision list. They extracted semantic and grammatical features from the context of members of confusion set using corpora. Chaudhuri [5] described a new novel technique of location and correction of non-word error. They pinpointed the error position in a big majority of cases and thus reduce the number of correct alternatives to a large extent. Their approach was based on matching the string in the normal as well as a reserved dictionary. They combined with a phonetic similarity key based approach where phonetically similar characters were mapped into a single symbol and a nearly phonetic dictionary was formed.

In this paper, we propose a Myanmar spell checker system for handing Myanmar spell errors by applying Myanmar Text Corpus, Levenshtein Distance Algorithm and Naïve Bayesian Classifier.

3. Nature and Collation of Myanmar Language

Myanmar language is a very rich language and use as an official language of the Union of Myanmar. A Myanmar syllable has a base character, and may also have (or not) a pre-base character, a post-base character, an above-base character and a below-base character. Syllables have to be constructed. Each syllable boundary should begin with a base consonant. Myanmar languages have 33 consonants and the consonant combines with vowel and sometime it includes medial to form the complete syllables in Myanmar language. Besides, it has not delimiter between syllables and words. Myanmar words are collated being based on syllables. A Myanmar syllable encoded in Unicode can be broken into 5 parts for collation [9]: <consonants> <vowels> <medial> <final> <tone>. In particular sentence, Typographic Errors (Non word errors) and Cognitive Errors (Phonetic Errors) are collocated with two or more syllables. But Context Errors (Real word errors) are only one syllable, which are ambiguous for poor reader.

The Myanmar saying “the pronunciation is merely the sound, whilst the orthography is correct” (ရေးတော့အမှန်၊ ဖတ်တော့အသံ) reflects the differences between spoken and written Myanmar, as spelling is often not an accurate reflection of pronunciation. Some writers are writing with the pronunciation and careless of spell error. In Myanmar Language, every isolated word has meaning. And also there have compound words. But some words are cannot combine as a compound word. If we combine the two words, the compound word’s meaning will be changed. For example, (စိမ်း→green) (လန်း→fresh). If we combine these two words their meaning will be changed as (စိမ်းလန်း → green and lush). Typist may misuse the word (လန်း) with (လမ်း→road). The two words (လန်း and လမ်း) have same pronunciation but different meanings. There is no combination of (စိမ်း and လမ်း). Myanmar words collocation depends on the previous meaning of words. One word has different meanings and different usages. So spell checker is major issue and challenge for all computerized applications of Myanmar Language. Myanmar syllables and defined symbols are shown in Table 1.

Table 1: Type of Myanmar Syllables and Defined Symbols

Syllables	Type of Syllables	Defined Symbols
က–အ	Consonants	C
◌ျ ၊ ◌ြ	Medials	M1
◌ွ ၊ ◌ှ	Medials	M2
ဧ	Vowels	V1
◌ာ၊◌ိ ၊◌ီ၊◌ု ၊◌ူ၊◌ေ ၊◌ဲ	Vowels	V2
◌်	Final	F
◌့ ၊◌း	Tone	T

Common misused characters and sample words are shown in Table 2.

Table 2: Common Misused Characters and Sample Words

Common Misused Characters	Sample Words
က ၊ ခ ၊ ဂ	ကမူ၊ ဂမူ ၊ ခုံးတံတား၊ ဂုံးတံတား
စ ၊ ဆ ၊ ဇ ၊ ဈ	ဇရက် ၊ ဆက်ရက်၊ စား၊ ဆား
ဏ ၊ န	အနုမြူ၊ အဏုမြူ
ပ ၊ ဖ ၊ ဘ ၊ ဗ	ဖူး၊ ဘူး၊ ဗူး ၊ ပုန်း၊ ဘုန်း
သ ၊ တ	သုံး ၊တုံး
ဒ ၊ ဓ	ဓါး ၊ ဒါး
ယ ၊ ရ	ယက် ၊ ရက်
◌ျ ၊ ◌ြ	ကြား ၊ကျား
တ် ၊ က် ၊ ပ်	တတ် ၊တက်၊ တပ်

နှ ၊ မ်	လမ်း ၊လန်း

Sample of compound misused words and correct words are shown in Table 3.

Table 3: Sample of Correct Words and Misused Words

Correct words	Misused words
ကတ်ကြေး	ကပ်ကြေး
ခြေလှမ်း	ခြေလှန်း
နေလှန်း	နေလှမ်း
ထွန်ယက်	ထွန်ရက်
ကွန်ရက်	ကွန်ယက်
ဆက်ရက်	ဇရက်
ချီတက်	ချီတတ်
စိမ်းလန်း	စိမ်းလမ်း
စွဲလမ်း	စွဲလန်း

4. Types of Spell Errors

Spelling error patterns can result generally from the mistakes made by human. Generally, human-generated misspellings can be distinguished into four groups: (i) Typographic Errors (Non-word errors) (ii) Sequence Errors (iii) Phonetic Errors (Cognitive errors) and (iv) Context Errors (Real word errors)

Typographic Errors: These errors have been made by the typist accidentally presses the wrong key, presses the keys in the wrong order (e.g., misspelling 'edit' as 'adit'). These errors are made assuming that the writer or typist knows how to spell the word but may have typed the word hastily resulting in an error. For example, "သူ**ကျောင်**သို့သွားသည်" . In this sentence, mistyped word is (ကျောင်). The typist need to type (း, tone). The word (ကျောင်) has no meaning and for the above sentence the correct word is (ကျောင်း→school). မိုးခေါင်သောကြောင့် ကုန်စျေး**နှုံး** များ မြင့်တက် လာသည်။ In this sentence, the word **(နှုံး)** has meaningless. The correct word is (**နှုန်း**→price).

Sequence Errors: These errors can be caused in writing Myanmar words with wrong format sequence that may be two or three combinations of consonants, medial or vowels. For example, (pigeon, ခို →"ခ– ◌ိ– ◌ု" as "ခ– ◌ု ◌ိ") (two combinations) and (dove, ချိုး→ "ခ– ◌ျ –◌ိ–◌ု– း" as "ခ ◌ိ ◌ျ – ◌ု – း") or ("ခ– ◌ျ –◌ိ –◌ု –း" as "ခ ◌ျ –◌ု ◌ိ – း") (three combinations).

Phonetic Errors (Cognitive Errors): They have been made by a lack of knowledge of the writer (e.g., misspelling 'separate' as 'saparate'). These errors are made when the writer substituted letters they believe sound correct into a word, which in fact leads to a misspelling where the misspelling is pronounced the same as the intended word but the spelling is wrong which accidentally produce a real word (e.g., misspelling 'weather' as 'whether').

Examples: (i) သစ်ပင်များ **စိမ်းလမ်း** စိုပြည် သည်။
(ii) ငါးများ ကို **နေလှမ်း** သည်။
(iii) ကွန်ပျူတာ **ကွန်ယက်** ဆက်သွယ်ရေး စနစ် သည် လွန်စွာ အသုံးဝင် သည်။

In example sentence (iii), there has no combination of (ကွန်ယက်). ". The two words (**ယက်** → rake) and (**ရက်** → weave) have the same pronunciation but different meaning. The correct combination for those error words is (ကွန်ရက်→network). The correct sentence is ကွန်ပျူတာ **ကွန်ရက်** ဆက်သွယ်ရေး စနစ် သည် လွန်စွာ အသုံးဝင် သည်။ "Computer network system is very useful".

Context Errors: They can be seen to be a subset of phonetic errors which produce a real word error (e.g., misspelling 'piece' as 'peace'), where the word is pronounced the same as the intended word (e.g., 'dessert' as 'desert') but the word is ambiguous for the input sentence.

Examples*:* (i) ရာသီဥတုသည် တော ကို **မီ** သည်။
(ii) **မျှား** အတွင်း ကား မရပ်ရ။
(iii) ငှက်များ လေထဲ တွင် **ပြန်** နေကြသည်။

In example sentence (iii), there has ambiguous word (ပြန်). Some writer misused the context word like that the word (ပြန် →return) is used. The correct word for that sentence is (ပျံ →fly): ငှက်များ လေထဲ တွင် **ပျံ** နေကြသည်။ "Birds are flying in the air". In Myanmar words, (ပြန် and ပျံ) have the same pronunciation but difference meanings and difference usages.

5. Myanmar Spell Checker Framework

Myanmar Spell Checker Framework consists of four components as shown in Figure 1. They are: (1) Myanmar Text Corpus, (2) Tokenizer, (3) Spell Checker and (4) Suggestion Generator.

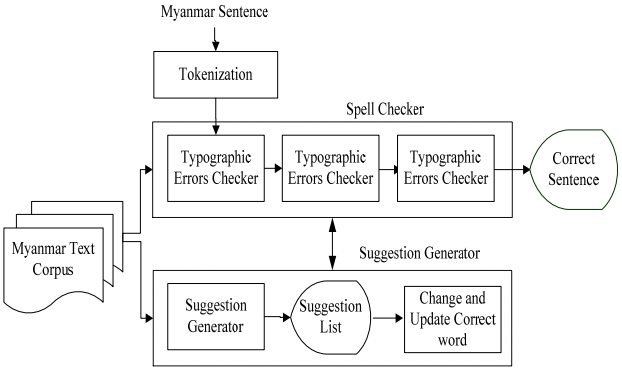

Figure 1. Myanmar Spell Checker Framework

5.1 Corpus Creation

Corpus is a large and structured set of texts. It is used to spell checker, checking occurrences or validating linguistic rules on a specific universe. Besides it is a fundamental basis of many researches in NLP. Building of the text corpus is very helpful for the development of spell checking. In this work, Myanmar text corpus is created manually to apply in Myanmar Spell Checker system. It contains various sense meanings of ambiguous Myanmar words, compound words and training sentences. All words are collected from example sentences of "Myanmar Grammar" [10], "Myanmar Words Commonly misspelled and misused books [7]", "Ornagai Dictionary" [16] and "Wxpy Dictionary" [17].

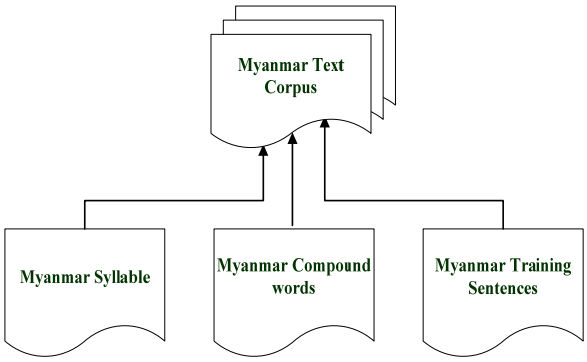

Figure 2. Myanmar Corpus Structure

Myanmar Syllable file is used for checking Typographic errors which consists of 1908 syllables. Myanmar Compound Words files is used for checking compound misused errors which misused as phonetic errors, it also used for segmented words for the input string. In Myanmar Compound words file, which consists of 62582 compound words. Myanmar Training sentences consists of 3600 sentences and average words in sentences is 12. Training sentences are used for calculating the probabilities of Context words errors.

5.2 Tokenization

Tokenization is a preprocessing step for this system. It is the process of breaking a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. The list of tokens becomes input for further processing such as parsing or text mining. Tokenization is useful both in linguistics and in computer science, where it forms part of lexical analysis. Typically, tokenization occurs at the word level. However, it is sometimes difficult to define all contiguous strings of alphabetic characters and to define what is mean by a "word". Tokens are separated by whitespace characters, such as a space or line break, or by punctuation characters. In languages such as English where words are delimited by whitespace, this approach is straightforward [19]. However, tokenization is more difficult for languages such as Myanmar, Thai, Japanese, and Chinese which have no word boundaries.

Myanmar text is a string of characters without explicit word boundary, so it is hard to define word boundary. In this paper, we describe regular expression and pattern for tokenization of a word boundary with Finite State Automata. An automaton can be said to recognize a string [1]. In Myanmar3, start state is always started with Consonant (C) and "end sate" is represented with double circle. Each character in the input string passes through the corresponding edges to the next state. In this way, it reaches the final state, and then automatons accept the input string and return a word with boundary. According to the Myanmar word collation rule (e.g., ကျောင်း = က + ျ + ော + င + ် + း < C M1 V1 V2 C F T >), we define the Finite State Automata in figure 3.

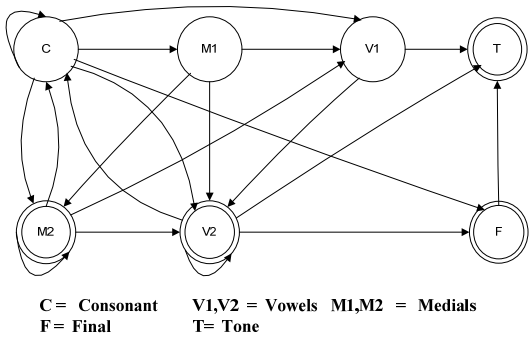

Figure 3. Finite State Automata for Tokenization of Myanmar Syllable

Examples of Myanmar Syllable collation

ကျောင်း	= က + ျ + ော + င + ် + း	<C M1 V1 V2 C F T>
လျှင်	= လ + ျ + ှ + င + ်	<C M1 M2 C F>
မြန်	= မ + ြ + န + ်	<C M1 C F>
ကောက်	= က + ော + က + ်	< C V1 V2 C F >
စိုက်	= စ + ို + က + ်	< C V2 V2 C F>

6. Implementation of Myanmar Spell Checker

6.1 Detection of Typographic Errors

Non-word errors correction is an important task. Non-word error spelling correction is focused on the task of generating and ranking a list of possible spelling corrections for each words not existing in the corpus. It is also isolated words errors checking and generating suggestion. The main steps of Typographic Errors checking process are:

1. Look up the word in the corpus
2. In case, the word exit, pass on to next word.
3. If the word is not found in the corpus, calculate the similarity of the error words and word from corpus to generate suggestion list.

6.2 Detection Phonetic Errors

Phonetic error is a special class of real words errors in which the writer substitutes a phonetically correct but orthographically incorrect sequence of letter for the intended words. Moreover, there exists a class of real word errors in which the misspellings result in a valid word. It occurs due to the presence of words in the language having similar pronunciation but different meaning. In this paper, we

proposed Myanmar compound misused words detection algorithm which detect phonetic errors. To detect this type of errors, the system used proposed algorithm and bigram model. Then generate suggestion list by applying Levenshtein Distance Algorithm.

6.2.1 Longest Matching approach and Proposed Algorithm for compound misused words

In Myanmar language, the text is a string of character written in sequence from left to right and word are not always delimited by spaces although sentences are clearly delimited by a sentence boundary marker "။". The text is needed to be segmented into words as the preprocessing step in order to process phonetic error checking. Many methods for word segmentation have been proposed. We apply with longest matching approach to check phonetic error of Myanmar compound words. Syllable level longest matching algorithm is described by [15].

After word segmentation, we check the segmented words with the proposed algorithm to check phonetic errors by applying corpus lookup approach. The proposed algorithm is shown in figure 4. To detect compound misused words, we calculate the probability of next word by using bigram model depict in Equation 1.

$$P(W_n \mid W_{n-1}) = \frac{P(W_{n-1}, W_n)}{P(W_{n-1})} \quad (1)$$

If there have probability, we define as correct compound word. If there have no probability we define as misused compound words and calculate similarity of the two continuous words.

```
1. Input  : Segmented words
            Seg←length of the segmented words
            SW←predefined stop word list
            CW ←predefined context words
            int j←1
2. while(Seg>0)
3.  {       w1←Seg(j-1);
4.           w2←Seg(j);
5.           w3←Seg(j+1)
6.
     if w1and w2 are equal with "Oneword" and not contain
     in SW  and CW
7.   {
8.            Perror←w1+w2;

9.      if Perror not contain in the bigram word list
10.    {
         If w3  not contain in the CW and SW and equal with
                              "Oneword"
11.       {   W4←Perror+w3;
12.          Calculate Similarity of w4 with word from
     corpus;
13.     }
14.     else
             Calculate similarity of Perror with word from
     corpus
15.  }
16.  else if   w1 equal with  "Twoword" and w2 equals with
     "Oneword"
17.  {
     If w2 not contain in the CW  and SW
         {
18.   cmbW←w1+w2;
19.        If cmbW not contain in the bigram  word list
20.      {
21.      If w2 and w3 not contains in the CW and SW and
     w3 equals with "Oneword"
22.         {
23.          Perror←w2+w3;
24.         If Perror not contain in the bigram word list
25.            Calculate similarity of Perror with word from
     corpus;
26.         else Calculate similarity of cmbW with word
                 from corpus;
27.         }
28.      }
29.    }
30.   }
31. }
32. else if  w1 equal with "Oneword" and w2 equals with
     "Twoword"
33.  {
34.   If w1 not contain CW and SW
35.   {
36.      cmbW←w1+w2;
37.      If cmbW not contain in the bigram word list
38.     {
39.      Calculate similarity of cmbW with word from
     corpus;
40.      }
41.    }
42.   } end if
43.   Seg←Seg-j
44.  }end while
45. Print correct words
```

Figure 4. Proposed Algorithm for Detection of Compound Misused Words

6.3 Suggestion Generator

After checking the whole sentence, the system detects error words. If error words present in the input sentence, it calculate the similarity of the error word and word from corpus. And then we generate the possible suggestion list which rank according to the most likely candidate by using Collection sort. The errors words and correct words will have a Levenshtein distance less than or equal to 3 which are considered to get more similar Myanmar words.

6.3.1 Levenshtein Distance Algorithm

There are many kinds of String Similarity Algorithms for spelling checking such as Hamming Distance, N-grams, Longest Common Subsequence (LCS), Cosine Similarity and Levenshtein Distance (LD). Among these algorithms, Levenshtein Distance Algorithm is the best algorithm for two fuzzy strings. In information theory and computer science, the LD is a metric for measuring the amount of difference between two sequences (i.e., the so called edit distance). A generalization of the LD allows the transposition of two characters as an operation and produces the number of operations need to be transformed from one word to another. LD is a measure of the similarity between two strings, which we will refer to as the source string (s) and the target string (t). It is used in some spell checkers to guess at which word (from a corpus) is meant when a missed

spelt word is encountered and operate Insert, Delete and Substitute transformations. At the end, the bottom-right element of the array contains the answer. The resulted distance is the number of deletions, insertions, or substitutions required to transform ***s*** into ***t*** [18].

6.4 Detection and Suggestion Generation of Context Errors

In Myanmar Language, most of context words are Myanmar verb. For example, the confusion set {မှီ၊ မီ} have the same pronunciation but difference meaning. In the context word "မှီ" which would translate to English word "base on some fact or evidence", "မီ" which use to combine with other Myanmar Noun and verb. For example: "be with (reach, time, limit) (as in လက်လှမ်းမီ၊ အချိန်မီ) "and then it can use as part :before; prior to (as in မသွားမီ). Confusion set consists of words that are likely to be misused in place of one another. We can see in the following sentence that misused " မှီ " instead of " မီ ". သူသည်ဘူတာရုံသို့အချိန်မှီလာသည်။" The correct word for that sentence is " He come railway station on time".

In Myanmar Language, all context words can correct by statistical techniques exception for {ဖူး၊ဘူး}{ဘဲ ၊ပဲ} confusion set. In Myanmar words, (ဘူး and ဘဲ) are always use as negative statement. The two words always combine with (မ), for example: မရေးဘူး။ (not write), မစားဘဲနေသည်။ (live without eating) . Myanmar verb always use between မ and ဘူး / ဘဲ for describe negative statement. Myanmar context words (confusion set) are shown in table 4.

Table 4: Myanmar Context Words

Confusion set	
ပဲ	ဘဲ
ဖူး	ဘူး
ဖက်	ဘက်
မှီ	မီ
မျှား	မြား
မင်	မှင်
ပျံ	ပြန်

In the literature, the problem of context sensitive spelling correction is commonly formulated as a disambiguation task where ambiguity among words is model by confusion sets. A confusion set C= {W_1, W_2,……,W_n} means that each words W_i in the set is ambiguous with each other words in the set. Thus, if C={piece, peace} and either piece or peace encountered in a text, the task is to decide which one was intended. This way of identifying the actual form from an observed or surface form is called Bayesian classification.

Figure 5 shows the process of context errors detection and suggestion generation by using Naïve Bayesian classifier. The idea of the Naïve Bayesian classifier which we will present for word senses is that it looks at the words around confusion set in a large context window. Each content word contributes potentially useful information about which sense of the ambiguous word is likely to be used with it. The supervised training of the classifier assumes that we have a corpus where each use of ambiguous words is labeled with its correct sense. For context error detection and correction tasks, giving a word w, candidate classification variables S=(S_1,S_2,….,S_k) that represent the sense of the ambiguous word and the feature F=(f_2,f_2,….,f_n) by that describe the context in which an ambiguous word occurs, the Naïve Bayesian finds the proper sense s for the ambiguous word w by selecting the sense that maximizes the conditional probability P(w=s_i|F). Suppose C is the context of the target word w, and F=(f_1,f_2,…,f_n) is the set of features extracted from context C, to find the right sense s! of w given context C, we have:

$$s' = \arg\max\left[\sum_{f_j \in C} \log P(f_j \mid w = s_i) + \log P(w = s_i)\right] \quad (2)$$

S = (S_1,S_2,….,S_k) sense of the context words

F = (f_1,f_2,….,f_n) , the set of features extracted from sentence which an confusion word occurs,

P(s_i) = The probability of sense (ambiguous word) s_i

P(f_j|s_i)= the conditional probability of feature f_j with observation of sense s_i

The probability of sense s_i, P (s_i), and the conditional probability of feature f_j with observation of sense s_i, P(f_j|s_i), are computed via Maximum-Likelihood Estimation:

$$P(s_i) = C(s_i) / C(w) \quad (3)$$

$$P(f_j \mid w = s_i) = C(f_j, s_i) / C(s_i) \quad (4)$$

Where C(f_j,s_i) is the number of occurrences of f_j in a context of sense s_i in the training corpus, and C(s_i) is the number of occurrences of si in the training corpus, and C(*w*) is the total number of occurrences of the ambiguous word *w* [4]. To avoid the effects of zero counts when estimating the conditional probabilities of the model , when meeting a new feature f_j in a context of the test dataset, for each sense s_i, we set P(f_j|w=s_i) equal 1/C(w).

i=position;
Step 1: Processing
- a. Segment input sentence by longest matching
- b. Remove stop words from input

Step 2: Confusion set Lookup
- a. Lookup possible sense from the corpus Confusion word equal with { ဖူး၊ဘူး၊ဘဲ ၊ပဲ} go to step(3)
- b. if not equal go to step(4)

Step3: Confusion word equal with { ဖူး၊ဘူး၊ ဘဲ၊ပဲ }

If sense equal with(ဘဲ) and position of sense equal 1
Change (ပဲ);

else if sense equal with (ဘူး) and position of sense equal 1
Change (ဖူး);

else if sense equal with (ဖူး or ပဲ) and sense position 0 or 1 equal with မ
Change (ဘူး or ဘဲ);

-else if sense equal (ာ္ or ာ်) and sense position 0 or 1 is not equal with ေ
Change (ာ္ or ာ်);
-else
print correct;

Step4: a) Calculate Probability
-for all sense s_i of W do
-for all words fi in the vocabulary do
$P(f_i| s_i)= C(f_i,s_i)/ C(s_i)$
-end
-end
-for all senses s_i of W do
$P(s_i)= C(s_i)/ C(w)$
-end

b) Disambiguation
-for all sense s_i of W do
-score $(s_i)=\log P(s_i)$
-for(all words f_i in the context window c do
-score $(s_i)=score(s_i)+\log P(f_i|s_i)$
-end
-end
Choose s'=arg max score(s_i)

Figure 5. Process of Detection and Suggestion Generation of Myanmar Context Errors

7. Experimental Results

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). Testing sentences are used for evaluation which consists of words include in corpus, test sentence that are not exactly same sentences in corpus, and new words. Corpus size is larger and larger because the tested sentences are manually added to the corpus to get accuracy for new words which are not included in corpus.

In this system, we tested with 500 sentences to get the accuracy of the system. The average numbers of words includes in one sentence is 12 words. Figure 6 shows the accuracy of correctly detected on the testing sentences with the compound words errors detected algorithm. Figure 7 shows similarity score suggestion generation for Typographic errors and Phonetic errors by using Levenshtein Distance Algorithm. In that figure, suggestion generation of Typographic errors get 100% accuracy. But, at the Phonetic errors, 91% similarity score of suggestion list are generated. Table 5 shows the accuracy of context errors detection and suggestion generation results. Average accuracy of overall system gets 95% precision, 92.33% recall and 93% f-score.

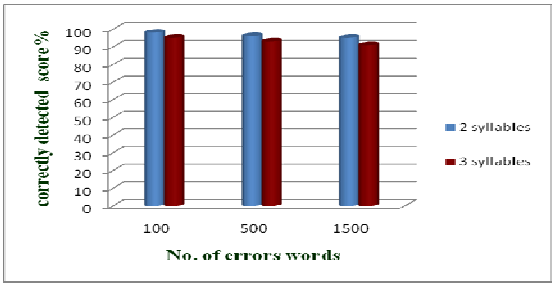

Figure 6. Experimental result of Phonetic Errors Detection

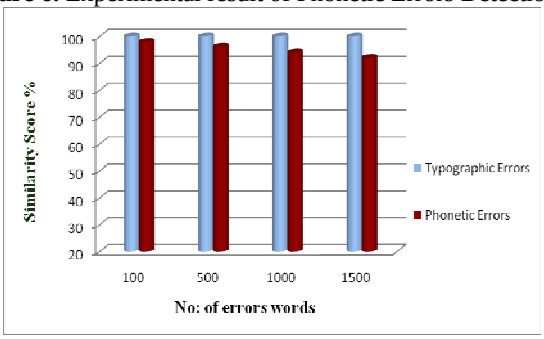

Figure 7. Similarity Score of Suggestion Generation for Typgraphic Errors and Phonetic Errors base on Levenshtein Distance Algorithm

Sentence Types	Accuracy (%)
Test sentence in the corpus	98%
Test sentence that are partial words include in the corpus	91%
Test sentence that are not include in the corpus	82%

Table 5. Context Errors Detection and Suggestion Generation Results

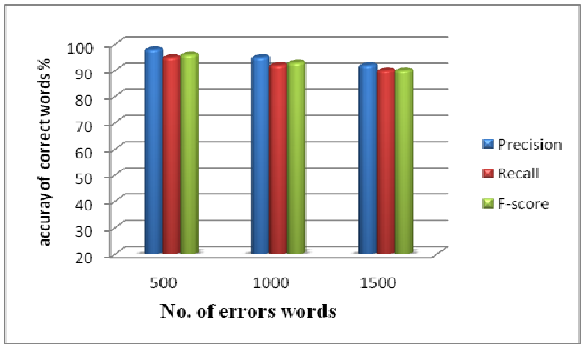

Figure 8. OverAll System Evalutaion Results on Accuray of Correct words Vs. No. of Sentences

8. Conclusion

We implemented a spelling checker system for Myanmar language which can handle Typographic errors, Sequence errors, Phonetic errors and Context errors. A Myanmar Text Corpus is created and Mynamar3 Unicode is applied for implementing the Myanmar Spell Checker system. We applied Levenshtein Distance Algorithm, for generating

suggestion list. The proposed algorithm is very useful in checking compound misused word 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. Evaluation results show that this system can provide promising accuracy.

References

[1] Lewis II, P.M, Rosenkrantz, D.J, Stearns, R.E, Compiler Design Theory, Addison_Wesley Publishing Company, Third Printing November, 1978.

[2] M. Nagata, Context-Based Spelling Correction for Japanese OCR, Proceeding of the 16th International Conference on Computational Linguistics, 05-09, 1996.

[3] Golding, A, A Bayesian hybrid method for context sensitive spelling correction, In Proceeding of 3rd workshop on Vary Large Corpora, 3, Jun 1996.

[4] Christopher D. Manning, Foundations of Statistical Natural Language Processing, The MIT Press, Cambridge, Massachusetts, London, England, 1999

[5] B. Baran Chaudhuri, Reversed word dictionary and phonetically similar word grouping based spell-checker to Bangla text, LESAL workshop, 2001.

[6] T Dhanabalan, Ranjani Parthasarathi, T V Geetha, Tamil Spell Checker, Resource Center for India Language Technology Solutions, TDIL newsletter, 2003,Tamilnadu, India.

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

[8] N. UzZaman, M. Khan, A Bangla Phonetic Encoding for Better Spelling Suggestion, Center for Research on Bangla Language Processing, Proceeding of 7th International Conference on Computer and Information Technology, Dhaka, Bangladesh, Dec, 2004.

[9] Stribley, K, Collation of Myanmar in Unicode, 22, August 2005

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

[11] N. UzZaman, M. Khan, A Comprehensive Bangla Spelling Checker, Center for Research on Bangla Language Processing, Proceeding of International Conference on Computer Processing on Bangla, 2006.

[12] မြန်မာစာ မြန်မာ စကား , Department of Myanmar Language commission, Ministry of education, Union of Myanmar June 2007.

[13] Md. Munshi Abdullah, Md. Zahurul Islam, Mumit Khan, Error tolerant Finite State Recognizer and String Pattern Similarity Based Spelling Checker for Bangla, Proceeding of 5th International Conference on Natural Language Processing (ICON), 2007.

[14] D. Fossati, B. D. Eugenio, A Mixed Trigrams Approach for Context Sensitive Spell Checking, Proceeding of the 8th International Conference on Computational Linguistics and Intelligence Text, 2007.

[15] H.H. Htay, K. N. Murthy, Myanmar Word Segmentation using Syllable Level Longest Matching, Proceeding of the 6th Workshop on Asian Language Resources, 2008.

[16] http://www.ornagai dictionary.com

[17] http://www.Wxpy dictionary.com

[18] http://www.Enclyclopedia.com/ Natural language understanding / Levenshtein_distance.html

[19] http://www.wikipedia.com/Tokenization.hml

Author Profile

Aye Myat Mon is currently pursuing Ph.D degree program in University of Computer Studies, Mandalay, Myanmar. I got M.C.Sc from University of Computer Studies, Yangon in 2008. I am also an assistant lecturer. My current research interest is Natural Language Processing.
E-mail:amyatmon99@gmail.com

Thandar Thein received her M.Sc. (Computer Science) and Ph.D. (Information Technology) degrees in 1996 and 2004, respectively from University of Computer Studies, Yangon (UCSY), Myanmar. She did her post doctorate research fellowship in Computer Engineering Department of Korea Aerospace University, the scholarship awarded by Korea Research Foundation Grant funded by the Korean Government. She is a faculty member of UCSY since 1996. Currently she is a professor and guiding the Ph.D students. Her current research interests are in the areas of Software Aging, Virtualization, Green Computing, Wireless Sensor Networks and Natural Language Processing.
E-mail:thandartheinn@gmail.com