Abstract

SpellCheF is a spell checker for Filipino that uses a hybrid approach detecting and correcting misspelled words in a document. Its approach is composed of dictionary-lookup, n-gram analysis, Soundex and character distance measurements. It is a plug-in to OpenOffice Writer. This research involves incorporating into a spell checker system two spelling rules and guidelines, namely, the KWF 2001 Revision of the Alphabet and Guidelines in Spelling the Filipino Language, and the Gabay sa Editing sa Wikang Filipino rulebooks. The automation of the rules and guidelines helps the system to carry out its task. The learned rules allow the system to spell words when assimilated to Filipino and also check the spelling of a document written in Filipino. The system through the use of rules detects misspelled words in a given document and in return generates a list of possible suggestions for the user to choose from. The misspelled words can be updated with the correct word at once.

1. Introduction

A spell checker is a feature of a document-related or a stand-alone application. It is used in ensuring the correctness of the spelling. Simple spell checkers use word per word comparison. They utilize a large lexicon to compare words with. For words which are not found in the lexicon, algorithms are sometimes used in order to tell which word the user possibly wants to use.

[1] implemented a tri-gram-based Tagalog spell-checker. It was designed to eliminate the use of a dictionary to perform spell checking. It undergoes learning with a corpus of Tagalog text as input to build the tri-gram lookup table where the spell checking will be based. Spell checking goes through almost the same process with Tagalog text as input and checking is done not on the word itself but on the tri-gram it contains.

Filipino is the term used in 1973 and 1987 Philippine Constitutions to designate as the national language. It is sometimes referred to as Tagalog. But the two languages are different. Orthographically, Tagalog has 20 letters while Filipino has 28 (which include C, F, J, N, Q, V, X, and Z).

The Sentro ng Wikang Filipino of UP-Diliman and the Commission on Filipino Language came up with different guidelines on spelling Filipino words, called the Gabay sa Editing sa Wikang Filipino[5] and the Alfabeto at Patnubay sa Ispelling ng Wikang Filipino[2], respectively. Throughout this paper, we will refer [5] the Gabay guideline, while [2] will be referred as the KWF guideline. The two guidelines may have similarities, but there are also differences, specifically on the spelling of borrowed words. For instance, the KWF guideline has a rule on when to change the letter Q to K or to KW, while the Gabay guideline did not give any guidelines on how to change Q. Another difference is in the spelling of words with letter X. Xerox (meaning photocopy) would be spelled as seroks following the Gabay guideline, and xerox following the KWF guideline.
2. SpellCheF

SpellCheF stands for Spell Checker in Filipino. It is a spelling error detection and correction system for Filipino that utilizes a hybrid approach. Based on the spelling guidelines of the Komisyon ng Wikang Filipino[2] and/or Gabay sa Editing sa Wikang Filipino[5], it is able to detect spelling errors and give appropriate suggestions. The architectural design of SpellCheF is shown in Figure 4. The system is composed of four main parts, namely, the lexicon builder, n-gram analysis, detection, and correction modules. The lexicon builder is responsible for grouping the words from the corpora and creating the lexicon which will then undergo n-gram analysis. The Detector module is responsible for marking possibly misspelled words in a given document, while the Corrector module generates a list of suggestions on how to correct the detected misspelled word.

3. Lexicon Builder

The lexicon builder is used to extract words from source materials and categorize them according to the spelling guidelines which they follow. It creates three databases. The first one contains the KWF-compliant words. The second one stores the GABAY-compliant word. Finally, a third database stores the common words of KWF and GABAY. These three databases are used as dictionary by the Detector and Corrector modules. Table 1 summarizes the differences in the two spelling guidelines.

<table>
<thead>
<tr>
<th>Table 1. Comparison of the Spelling Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>KWF</td>
</tr>
<tr>
<td>Digraph SH</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Repetition with Clustered Consonants</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>“Diptonggo”</td>
</tr>
<tr>
<td>Ia</td>
</tr>
<tr>
<td>Ie</td>
</tr>
<tr>
<td>Io</td>
</tr>
</tbody>
</table>

Based on the differences in Table 1, rules were formulated to filter the words for a specific guideline. These rules are stored in a file and read by the Lexicon Builder. These rules are represented in regular expressions, using the java.util.regex API.

For example, in KWF digraph *sh*, word with *sh* or *sy* are both acceptable. The rule pattern would be `((\w*([s][hy][aeiou])\w*)`. \w* represents any character of any number. s may be followed by h or y, and then followed by the vowels a, e, i, o, u. Hence, the given pattern will accept words like *shopping, syaping, workshop, worksyap, bisyap, internsyip*.

4. n-gram Analysis

An n-gram as an $n$ letter subsequence of a string, where $n$ usually is 1, 2, or 3. In general, n-gram analysis techniques check each n-gram in an input string against a precompiled table of n-gram statistics to determine whether the n-gram can occur in a word. If it does, its frequency of occurrence in the words of the language is computed. Strings containing n-grams that do not occur in words or occur very infrequently are considered to be misspellings[3].

There are a variety of n-gram models, including the binary n-grams and the n-gram frequency statistics. In the n-gram table of the binary n-gram, its element has a value set to 1 or 0. While in the n-gram Frequency Statistics, the elements in its table has a value of its frequency counts or probabilities. These statistics are generated from a sufficiently large corpus of text covering a certain domain(s). [3] further classified the binary n-grams into positional and non-positional. In positional n-gram given a binary trigram array, the $i, j, k$th element would have a value 1 if and only if there exists one word in the lexicon with letter $l, m, n$ in positions $i, j, k$. While a non-positional n-gram, the position of the n-gram is not relevant within a word. SpellCheF implements non-positional tri-gram analysis. The tri-gram analysis module generates three n-gram look-up tables – one for the KWF lexicon, another for the Gabay lexicon, and one for the Common Words lexicon.

---

1 http://java.sun.com/docs/books/tutorial/essential/regex/index.html
5. The Detector Module

The Detector Module is responsible in determining if a word is considered misspelled or not. It is invoked through OpenOffice Writer. The user has to indicate whether the Detector should follow the KWF or GABAY spelling guidelines in detecting misspelled word.

To illustrate how this module works, consider the word bayano. This module will first check if bayano is in the lexicon. Since it is not in the lexicon, the system cannot automatically say that it is misspelled, for it may be a valid word but is not yet encountered by the Lexicon Builder. To determine if this word is possibly acceptable, the n-gram look-up table will be used. This method will check first if the length of bayano is greater than three to be able to generate tri-grams. The first extracted tri-gram is bay. If bay is in the n-gram look-up table, its frequency statistics is compared against the system’s threshold of 0.001. If it is less than the threshold, then it means that bayano is considered as misspelled. On the other hand, if the frequency statistics is greater than the threshold, the process is repeated for the next tri-gram which is aya. When all the word’s tri-grams’ frequency statistics exceed the threshold, the system considers the word as correctly spelled.

6. The Corrector Module

The Corrector Module is responsible in generating suggestions that would result to the possible correct spelling of an erroneous word. This module makes use of 2 different algorithms in generating suggestion: a similarity key algorithm that uses Soundex code and Tri-gram Analysis.

6.1 Suggestions using tri-gram analysis

The first step would be generating suggestions using the Tri-gram look-up table. The suggested words’ length is ± 1 character difference as the misspelled word. This constraint is based on studies which conclude that most misspelling involve at most 1 character change from the intended word [3].

The n-gram look-up table contains entries of the possible “grams” in the Filipino Language. Each gram entry also includes its frequency occurrence, possible previous gram and possible next gram. Two methods were used to generate the possible suggestions. The first method assumes that the first trigram of the misspelled word is correct. The second method assumes that the last tri-gram of the misspelled word is correct.

Let us consider the misspelled word panget. It is assumed that the pan gram is correct meaning the actual correct word starts with pan. The system will generate for words that start with pan like pangat, pangit, pantay.

The second method assumes that the starting characters are wrong, thus the last tri-gram is used to generate suggestions. For example, the misspelled word eboqasya, the generation of the suggestions will begin at the last gram aya, resulting to words like abogasya, ipinasaya, nagkasaya, pantasya.

6.2 Suggestions using Soundex

Soundex stands for “Indexing on Sound” [3]. This is a phonetic algorithm for indexing names by their sound when pronounced in English. This research applied Soundex to Filipino. Some Filipino words sound alike but are spelled differently. A Soundex code is created for similar-sounding letters. The idea of the code is to preserve, in a rough way, the salient features of the pronunciation. Vowel letters are discarded and consonant letters are grouped if they are likely to be substituted for each other. The Soundex algorithm is listed below:

1. Keep the first letter (in upper case)
2. Replace remaining characters using the Soundex values as shown in Table 2.
3. Delete adjacent repeats of a number
4. Delete the hyphens.
5. Keep the first three numbers or pad out with zeros

Consider the misspelled word imlpuwensya. Using Soundex, the suggestions is shown in Table 3:

<table>
<thead>
<tr>
<th>Table 3. Example of Generated Soundex Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>impluwensiya</td>
</tr>
<tr>
<td>i-level</td>
</tr>
<tr>
<td>i-rebuild</td>
</tr>
<tr>
<td>implikasyong</td>
</tr>
</tbody>
</table>

7. Ranking the Suggestions

After the Detector generates a list of correct spelling suggestions, the next step is to rank these suggestions. The character distance method is used to arrange and trim the suggestions. This method uses a Pythagorean-type metric to measure the distance between a misspelled word and a possible correction, based on the QWERTY keyboard layout[4]. The QWERTY keyboard is represented as a two 2-dimensional arrays: one for the lower case keys and another for the upper case keys. The suggested word with the shortest distance to the misspelled is the considered as the best suggestion.
Figure 2. Character Distance Chart

For example, the character distance between \( w \) with a location of (1,1) and \( n \) whose location (3,5) is \( \| (3,5)-(1,1) \| = \sqrt{2^2+4^2} \approx 4.47 \). Using this character distance, the scores for correction of imlpuwensya are (imlpuwensya; 2.83), (imlpuwensiya; 11.93), (imlpuwensyahan; 16.05).

8. Results And Recommendations

The system was tested using three Filipino documents:

1. An essay written by a student. The document is part of an introduction regarding the government under a known politician;
2. An entertainment article regarding an upcoming TV-series; and
3. A sports article about the Pacquiao – Solis match.

In testing this document, the dictionary had a total of 45548 words, where 2557 words categorized under Gabay, 4082 words under KWF, 42797 words are under the common words category. The document was tested using the Gabay guideline as spelling preference. The acceptance threshold used was 0.001.

The Detector module achieved a 7% error rate. These means 7% of the words the system marked as misspelled are actually correctly spelled. These errors are attributed to the following:

- A small lexicon. The corpora used may be considered small and some words are not found. Due to this, probabilities of grams in the tri-gram look-up table are not that accurate. Resulting sometimes to marking an accepted word as misspelled or accepting a misspelled word.
- Low frequency of the tri-gram of the word in the lexicon, thus the threshold was not met. By increasing the size of the lexicon, the quality of the n-gram frequency statistics would also increase.
- Two-letter words that are not in the lexicon, such as pa, o, di, k. Tri-gram analysis is applied only to words with at least three characters.

On the other hand, the Corrector module achieved a 94% accuracy rate. All correctly-detected truly misspelled words were given correct suggestions by the system. The errors came from correctly-spelled words which the Detector marked as misspellings. It is important to note that for a misspelled word, the correct word does not always appear as the first word in the suggestion list.

The length of the misspelled word and the suggestion greatly affect the rank of the suggestion. Since the system computes the distances of the two words, it has larger probability that the suggested word with the same length as the misspelled word will appear at the top of the list.

Results also show that the Soundex approach generates more suggestions than tri-gram approach. For efficiency and speed concerns, tri-gram approach only generates words that have less than ten characters. Because of this constraint, the tri-gram approach generates fewer suggestions. However, the quality of the tri-gram suggestions is usually better than the Soundex suggestions. This is because Soundex always assumes that the first letter is correct whereas tri-gram can suggest words whose first character is not the same as the misspelled word. It is, thus, recommended that improvements in the tri-gram based Corrector be implemented in the future. These improvements may consider the use of a hash table or better memory management techniques.

References

[4] Min, K., Wilson, W., Moon, Y., Typographical and orthographical Spelling Error Correction The University of New South Wales, Australia.