Abstract: Edit distance calculation is a well-known criterion for string-to-string comparison. It seems to be an appropriate method for alphabetic languages like English and European languages. However, the conventional edit distance criterion is not the best one for agglutinative languages like Korean. The reason is supposed that two or more letter units make a Korean character which is called as a syllable. This mechanism of syllable construction in the Korean language causes an inefficiency of edit distance calculation. We explored a new edit distance method by consonant normalization and the normalization factor.

Key-Words: edit distance, Korean character, consonant normalization, normalization factor

1 Introduction

Edit distance is defined as a cost of unit operations that is required to transform a string to the other string in which those two strings become the same string. There are four unit operations: insertion, deletion, replacement, and transposition in which insertions and deletions have equal costs and replacements have twice the cost of an insertion [1,2].

Edit distance is widely used for string-to-string comparison in alphabetic languages like English and European languages. However, the conventional edit distance metric is not so good for agglutinative languages like Korean [3,4]. We investigated the reason and found that word construction mechanism should be considered for the edit distance calculation. In the Korean language, two or more letter units make a Korean character which is called a syllable. Open syllable consists of a consonant followed by a vowel, and close syllable consists of three letters, consonant + vowel + consonant. Furthermore, when we write a complex consonant or a complex vowel, two key-strokes are required. That is, a syllable is completed in a computer environment by two to five key-strokes from an input device like keyboard or mouse.

The input mechanism of complex consonants and complex vowels needs two key-strokes, but they make a letter in a Korean word. This is what makes it hard to set a specific guide-line to calculate the edit distance. Suppose that a complex vowel ‘ㅘ’(wa) is substituted by ‘ㅝ’(wo). There are two candidates of edit distance. Edit distance would be 1 because only a letter is different. However, it may be 0.5 because half of complex vowel is different from the key-stroke point of view, ‘ㅏ’ and ‘ㅓ’.

2 Related Works

There are several distance measures: Hamming, Levenshtein, Damerau-Levenshtein, and Jaro-Winkler distance. Hamming distance is a metric between two strings of equal length [5]. It is used in an information theory to count the number of substitutions while transferring a text string. Levenshtein distance is a string-to-string distance metric that is recursively defined as in equation (1), where distance measures are insertion, deletion, and substitution [6].

\[
\text{dist}_{\text{LA}}(i, j) = \begin{cases} 
0, & \text{if } i = j = 0 \\
1, & \text{if } j = 0 \text{ and } i > 0 \\
1, & \text{if } i = 0 \text{ and } j > 0 \\
\text{min} \left( \begin{array}{c} \text{dist}_{\text{LA}}(i-1, j) + 1 \\
\text{dist}_{\text{LA}}(i, j-1) + 1 \\
\text{dist}_{\text{LA}}(i-1, j-1) + [a_i \neq b_j] \\
\end{array} \right), & \text{else}
\end{cases}
\]

(1)

Damerau-Levenshtein distance is an extension of the Levenshtein distance by adding a new operation:
transposition [7]. Jaro-Winkler distance has been
designed and best suited for short strings such as
person names [8]. Equation (2) describes the Jaro
distance $d_j$ of two given strings $s_1$ and $s_2$, where $m$
is the number of matching characters and $t$ is half the
number of transpositions.

$$d_j = \begin{cases} 0 & \text{if } m=0 \\ \frac{1}{3} \left( \frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) & \text{otherwise} \end{cases}$$ (2)

### 3 Edit Distance for Korean Words

Korean word is a sequence of syllables which is
called as a Korean character. Korean syllable is a
combination of consonants(C) and vowels(V)
in which only $C_1V\bar{C}_2$ format \(^1\) is allowed. Korean
character can be considered to be very similar to
Chinese character in which it consists of one or
more basic letter units. However, the internal
structure of Korean character is wholly different
from Chinese character in that Korean character is
separated into letters of consonant and vowels.
General edit distance metric has been devised and
applied for a simple, linear, and one-dimensional
string like English words [9,10]. When we try to
apply this edit distance metric to the two-
dimensional Korean word system, there are three
possible ways according to the basic operations.

#### 3.1 Letter-based Edit Distance

Letter-based metric is just like to apply conventional
Levenshtein distance to a letter string. A word is
supposed to be a sequence of consonants or vowels
with no consideration of syllable construction. This
method ignores a syllable boundary so that a
distance calculation for misspelled or transposition
errors across the syllable boundaries are easily
handled.

Letter-based metric easily computes transposition
errors that cannot be handled in syllable-based
metric, but it does not consider the syllable
characteristics of the Korean language where each
cword is a sequence of well-formed syllables $CV$ or
$C_1V\bar{C}_2$. Syllable-based system naturally put a
limitation of consonant and vowel combination. So,
most typo errors are affected by the syllable
formation rules and edit distance metric should reflect
this kind of syllable-structured characteristics.

#### 3.2 Syllable-based Edit Distance

Korean word is defined as a sequence of syllables.
Syllable-based edit distance is to count the number
of different syllables in a string. This metric is
adequate for long strings like text-to-text similarity,
but it cannot be used for short strings because all the
distance score is 0 or 1 regardless of the similarity
of syllables. The difference between syllables ‘다’
and ‘답’ is the final consonant ‘다’ and ‘답’ which is
one third of the syllable. Syllable-based metric
imposed a critical problem when typo errors occur
on a syllable boundary.

#### 3.3 Hybrid Edit Distance

Roh(2010) points out the inefficiencies of syllable-
based and letter-based method [3]. Syllable-based
metric has a drawback in which one letter difference
and whole syllable difference are treated as the same
distance.\(^2\)

$$d_{sy}(공원, 고원) = d_{sy}(공원, 낙원) = 1$$

In the above example, $d_{sy}(공, 고)$ is the same
distance as $d_{sy}(공, 낙)$, though there are two
matching letters in ‘공’ and ‘고’, where there is no
matching letter in ‘공’ and ‘낙’.

Roh(2010) proposed a hybrid method of syllable
and letter-based metric. He defined a $\beta$-distance
for syllable-based metric and $\alpha$-distance for letter-
based metric. If there is one or two letters match
then $\alpha$-distance is applied, and $\beta$-distance is used
for whole syllable difference. In the letter-based
metric, three letter-distance 3$\alpha$ is the maximum in
one syllable.

### 4 Phoneme-based Edit Distance

Hybrid edit distance has been devised for the
syllable structure of the Korean language. This
method divides a distance metric into syllable
distance and letter distance. However, this hybrid
method did not completely solve the syllable-
boundary changes. In this paper, we propose a new
method that considers phonetic change rules of the
Korean language.

#### 4.1 Phoneme Normalization

Our new approach of Korean edit distance
calculation introduces a phonetic pronunciation. As

---

\(^1\) Final consonant $C_2$ is optional.

\(^2\) The syllable concept is merely the basics of the Korean
writing system. In other points of view, Korean words are
treated as a sequence of consonants and vowels.
a preprocessing of input strings, consonants can be normalized to a phoneme representative. In this approach, nine phonetic transformation rules in Lee(2008) are sequentially applied [11]. After then, letter-based distance metric is applied to the modified input strings. Table 1 shows phonetically equal consonants and their corresponding representatives.

<table>
<thead>
<tr>
<th>Consonant</th>
<th>Representative</th>
</tr>
</thead>
<tbody>
<tr>
<td>ㄱ,ㅋ,ㄲ</td>
<td>ㄱ</td>
</tr>
<tr>
<td>ㄷ,ㄸ,ㅌ</td>
<td>ㄷ</td>
</tr>
<tr>
<td>ㅂ,ㅃ,ㅍ</td>
<td>ㅂ</td>
</tr>
<tr>
<td>ㅅ,ㅆ</td>
<td>ㅅ</td>
</tr>
<tr>
<td>ㅈ,ㅉ,ㅊ</td>
<td>ㅈ</td>
</tr>
</tbody>
</table>

**4.2 Edit Distance Normalization**

Edit distance normalization is introduced for two reasons. First, word-length independent metric will get a better result than word-length dependent metric. Second, non-similar words are easily filtered out when there are lots of word pairs for edit distance calculation.

\[
N = \frac{\alpha}{\max(|w_1|, |w_2|)} \quad (3)
\]

where \(|w_i| = 3 \times \text{syl\_length}(w_i)\)

Word length \(|w_i|\) is a syllable length of the word multiplied by 3 in which the length of a syllable is defined as 3. The range of \(N\) is between 0 and 1.

Zero means that two word strings are equal and 1 means that they are wholly different. For the example word pairs in Fig. 1, distance normalization score of <국어, 숙어> is 0.16(=1/6) and that of <나무가지, 나뭇가지> is 0.08(=1/12). These distance values mean that long word-length pair is more similar than short word-length pair.

**5 The Experimentation**

Experiments on normalization effects between two similar word pairs have been conducted. Experimental data are 699 confusing word pairs that have been built by the Ministry of Culture and Tourism [12]. Non-similar word pairs have been automatically constructed by all the other words except the one in the similar word pair. That is, non-similar word pairs are 688 for each word.

**5.1 Threshold Value of Normalization**

Table 2 shows the necessity of the normalization effect. X-axis is the threshold value for similarity decision and y-axis is the percentage of word pairs according to the threshold value. If we make a decision that a normalization score is less than 0.4, then 93.56% of word pairs are determined to be the synonym word.

**Fig. 2. Normalization for similar word pairs**

Threshold value for similar or non-similar word pair is controversy that we should induce the best threshold value. Fig. 4 shows the result of comparing the accuracy and threshold value 0.5 has got the best result.\(^3\)

\(^3\) For non-similar 243951 word pairs, 242732 word pairs have been determined correctly.
5.2 Performance Evaluation

We performed an experimentation and the results are shown in Table 2. Threshold value of the proposed method was set to 0.5. The methods in Table 2 are as follows.

- SyIED: Syllable-based distance metric
- LetED: Letter-based distance metric
- HybED: Hybrid metric of syllable and letter-based
- PhoED: Phoneme-based distance metric
- NphoED: Normalized phoneme-based metric

Table 2 shows that phoneme-based and normalized phoneme-based metric got a better result than the other methods. Especially, in the distance 2 and 5, phoneme based metric with normalization are better than the other edit distance metrics.

Table 2. Edit distance for similar word pairs

<table>
<thead>
<tr>
<th>Dst.</th>
<th>SyIED</th>
<th>LetED</th>
<th>HybED</th>
<th>PhoED</th>
<th>NphoED</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>14.2</td>
<td>22.3</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>68.4</td>
<td>68.4</td>
<td>68.7</td>
<td>71.2</td>
</tr>
<tr>
<td>2</td>
<td>75.9</td>
<td>89.6</td>
<td>89.1</td>
<td>91.6</td>
<td>91.6</td>
</tr>
<tr>
<td>3</td>
<td>94.2</td>
<td>93.8</td>
<td>95.7</td>
<td>95.7</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>96.2</td>
<td>95.8</td>
<td>96.7</td>
<td>96.7</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>98.4</td>
<td>98.0</td>
<td>98.6</td>
<td>98.6</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>99.3</td>
<td>98.5</td>
<td>99.0</td>
<td>99.0</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.4</td>
<td>99.4</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>99.9</td>
<td>99.9</td>
<td>99.9</td>
<td>99.9</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td>100.0</td>
<td></td>
</tr>
</tbody>
</table>

NphoED metric got the best performance of 91.6% in an edit distance 3 and 98.6% in the edit distance 6.

For the edit distance 1 and 2, our method got excellent results.

6 Conclusion

We proposed a new method of edit distance calculation for the syllable-structured word similarity. This metric has been devised to improve the performance of the syllable-based and letter-based metrics for word similarities. Edit distance performance has been improved through the phonetic pronunciation rules and word-length normalization.

Experimental results show that phoneme-based metric got a better result compared to the methods of letter-based, syllable-based and hybrid distance. Phoneme-based metric is improved by applying a normalization method which is better than other methods both for similar word pairs and non-similar word pairs.

References:

