Content list available at ITC

Techno-Science Research Journal

Journal Homepage: www.itc.edu.kh

Techno-Science Research Journal

Improving Recognition Result Using Character Trigram for Khmer OCR

Seangmeng Long*

Department of Computer Science,
Institute of Technology of Cambodia, Russian Ferderation Blvd., P.O. Box 86, Phnom Penh, Cambodia.

Abstract: *The recognition phase of an Optical Character Recognition (OCR) system produces a ranked list of candidate characters, among which the top one is usally taken as recognition result without taking context into account. Recognition error occurs if the correct character is not at the top, which is mostly due to shape similarity between characters.In this paper we propose to use character trigram, which means that two previous characters are taken into account when choosing the character from the candidate list as recognition result for Khmer OCR.A text corpus of about 300 Mbytes is used to compute character trigrams. Using these trigrams, we test our approach on about 3000 characters. The result shows that this approach can correct about 30% of recognition errors.*

Keywords: KhmerOCR; Character N-Gram; Error Correction

1. INTRODUCTION

Optical character recognition (OCR) is useful in a wide range of applications, such as office automation and information retrieval system. However, OCR in Cambodia is still not widely used, partly there is only few researches on Khmer OCR and existing Khmer OCRs are not quite satisfactory in terms of accuracy and font coverage.Several research projects have focused on spelling correction for many types of errors including those from OCR (Kukich, 1992).Nevertheless, the strategy is slightly different from language to language, since the characteristic of each language is different.

Two characteristics of Khmer which make the task of spelling correction different and difficult from those of other languages are: (1) there is no explicit word boundary, and (2) characters are written in three levels; i.e., the middle, the upper and the lower levels.In order to solve the problem of OCR error correction, the first task is usually to detect error strings in the input sentence. For languages that have explicit word boundary such as English in which each word is separated from the others by white spaces, this task is comparatively simple. If the tokenized string is not found in the dictionary, it could be an error string or an unknown word. However, for the languages that have no explicit word boundary such as Chinese, Japanese and Khmer, this task is much more complicated. Even without errors from OCR, it is difficult to determine word boundary in these languages.

There is no attempt on existing Khmer OCR systems (Chey et al., 2006; Ing, 2009) to correct recognition errors. In this paper we propose to use character trigram to correct recognition errors in Khmer OCR systems.The recognition phase of OCR systems produce a list of candidate characters ranked by how close they are to the image to be recognized. The ranking of the top five candidate characters will becombined with character trigrams to choose the correct one. The character trigrams – probability of a character knowing two previous characters – will be computed from a text corpus.

The paper is organized as follows. Section 2 describe our approach with an application to our Khmer OCR system. The experiments and results are discussed on section 3. We conclude our finding in section 4.

*Coresponding authors:
E-mail: seangmeng@itc.edu.kh; Tel: +855-78-406-183;
Fax: +855-23-880-369

2. THE APPROACH

2.1 Problem Statement

The problem of improving recognition result of Khmer OCR can be stated as follows:

Let $C = \{c_i\}$ the list of candidate characters produced by recognition phase of OCR systems with $R = \{r_i\}$ their ranking.

Let *c'* and *c''* the recognition result of the two previous characters.

Let $P(c_i \mid c'c'')$ the probability of character c_i knowing that the two previous characters are *c'c''*.

The problem is to choose the correct character c_i from the list of candidate characters.

Our approach is to convert the ranking into probability $P(c_i)$. As long as the recognition module is concerned, $P(c_i)$ is the probability that c_i is the correct character. So the problem becomes like this: finding c_i which maximize the value:

$$a*P(c_i) + b*P(c_i \mid c'c'') \quad \text{(Eq. 1)}$$

where *a* and *b* are weights we give to the two probabities $P(c_i)$ and $P(c_i \mid c'c'')$. Finding the good value for *a* and *b* is a challenging task.

The following sections will describe how to convert rankinig into probability, how to compute character trigrams and how to apply this approach to our Khmer OCR system.

2.2 Conversion of Ranking into Probability

Recall that r_i is the ranking of c_i. If higher value of r_i means higher ranking, $P(c_i)$ can be computed with this formular:

$$P(c_i) = r_i / Sum(r_i) \quad \text{(Eq. 2)}$$

where $Sum(r_i)$ is the sum of all r_i.

However if lower value of r_i means higer ranking, $P(c_i)$ can be calculated using this formular:

$$P(c_i) = (1/r_i) / Sum(1/r_i) \quad \text{(Eq. 3)}$$

2.3 Computation of Character Trigrams

The character trigrams are computed from a text corpus. They are calculated using this formular:

$$P(c_i \mid c'c'') = C(c'c''c_i) / C(c'c'') \quad \text{(Eq. 4)}$$

where:

- $C(c'c''c_i)$ is the number of occurrence of the sequence of characters $c'c''c_i$.
- $C(c'c'')$ is the number of occurrence of the sequence of characters $c'c''$.

When computing character trigrams, data sparseness issues must be taken into account. Data sparseness is a very serious and frequently occurring problem since the size of the corpus never seems to get enough. It means that if there is unseen trigram, the probability of the sequence is zero due to the equation above. The smoothing techniques make the distribution more uniform and redistribute probability mass from higher to lower probabilities. In this research, Laplace smoothing technique is selected to handle the issue. The basic idea of Laplace technique is to add one to every count. The formular becomes:

$$P(c_i \mid c'c'') = (C(c'c''c_i) + 1) / (C(c'c'') + B) \quad \text{(Eq. 5)}$$

Where *B* is the total number of characters.

2.4 Application to Khmer OCR System

We apply this approach to our Khmer OCR system by enhancing the recognition module with character trigram. This approach cannot be applied to our OCR system as is because Khmer word writing is not always in the same order as visually seen. For example, the sequence of characters "ក" and "ែ" renders as "ែក". The character "ែ" is written after "ក" but is place before "ក" when rendering. Moreover, while some characters produce different disconnected parts when rendering other sequence of two characters produce only one connected component. For instance the character "ញ" has two connected components. The sequence of characters "ច" and "ា" renders as one connected component "ចា". We will refer to these connected components as *symbol*.

So instead of character trigrams, we use symbol trigrams. c_i will represent symbol not character. The recognition module of our system produces a list of candidate symbols ranked by the distance from candidate symbol to the image to be recognized. So lower value of r_i means higher ranking. We use the formularin (Eq. 3) to compute $P(c_i)$.

For symbol trigrams, we have to convert text corpus into correct sequence of symbols as visually seen to compute symbol trigrams. For example, for the text "ចាញ់" composed of sequences of the following characters "ច", "ា", "ញ" and "់" will be converted

into sequences of the following symbols "ចា", "់", "ញ" and "~".

3. EXPERIMENTS AND RESULTS

We use a text corpus of about 300 000 phases (238Mbytes) to compute symbol trigrams. We test our system on 3473 symbols. Our Khmer OCR system without symbol trigrams produces 273 incorrect symbols. When applying symbol trigrams incorrect symbols drop to 188. So there are 85 symbols corrected. The correction rate is 31.14%. See table 1 below for detail.

Table 1. Experiment results

Total test symbols	3473

	Incorrect symbols	**Accuracy**
OCR without trigrams	273	91.88%
OCR with trigrams	188	94.39%

Total errors	273
Symbols corrected	85 (273 – 188)
Correction rate	31.14%

The result above is obtained using the value of coefficients $a = 0.8$ and $b = 0.2$. Recall that a is weight for $P(c_i)$ and b is for $P(c_i \mid c'c'')$. We give more weight to $P(c_i)$ because $P(c_i)$ is calculated from the real symbol image to be recognized. As for $P(c_i \mid c'c'')$ we give less weight as it is calculated from a text corpus which may not reflect well the actual text image to be recognized.

If we increase the weight of $P(c_i)$ there will be less corrections meaning that most of the time it will choose the first candidate character. At the other hand, if we increasethe weight of $P(c_i \mid c'c'')$ there will be more corrections which could lead to more miscorrections which mean that choosing the incorrect character while the first candidate is the correct one. We can try to improve the correction rate by tuning the coefficients *a*and*b*.

For the values we chose for $a=0.8$ and $b=0.2$, we noticed that there were only few miscorrections because a is much bigger than b.

As for list of candidate characters, we take only the top five candidates. While taking more candidate characters into account may lead to miscorrections, taking less candidate characters may lead to incorrect result as the correct character is not among these candidate characters. In our case, we noticed there are few cases where the correct character is not among the candidate characters.

4. CONCLUSIONS

We have explained the process to improve recognition result using character trigrams for Khmer OCR systems. The experimental result shows that our approach can correct about 30% of recognition errors. We may try to improve the correction rate by tuning a and b coefficients. In this work, we suppose that the two previous characters are already correctly recognized and try to choose the correct character from candidate lists using character trigrams. Another direction would be to try to find the correct phase given that we have list of candidate characters for each character in the phrase.

REFERENCES

Chey, C., Kosin, C. &Pinit, K. (2006). Khmer Printed Character Recognition by using Wavelet Descriptors. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems Vol.14 NO.3.337-350. Word Scientific Publishing Company.

Ing, L.I. (2009). Khmer OCR for Limon R1 Size 22 Report, PAN Localization Cambodia (PLC) of IDRC. URL: http://www.panl10n.net/english/Outputs%20Phase%202/CCs/Cambodia/MoEYS/Papers/2009/KhmerOCRLimonR122.pdf visited: october 2012.

Kukich, K. (1992). Techniques for automatically correction words in text.A CM Computing Surveys, 24(4).