Tamil to Hindi Machine Transliteration Using Support Vector Machines

S. Keerthana[1], V. Dhanalakshmi[2], M. Anand Kumar[1],
V.P. Ajith[1], and K.P. Soman[1]

[1] Computational Engineering and Networking,
Amrita Vishwa Vidyapeetham,
Ettimadai, Coimbatore
[2] Department of Tamil, SRM University, Kattankulathur
{keerthana.keerthi,dhanagiri,ajith12485}@gmail.com,
anandkumar@yahoo.co.in, kp_soman@amrita.edu

Abstract. Transliteration is the process of replacing the characters in one language with the corresponding phonetically equivalent characters of the other language. India is a language diversified country where people speak and understand many languages but does not know the script of some of these languages. Transliteration plays a major role in such cases. Transliteration has been a supporting tool in machine translation and cross language information retrieval systems as most of the proper nouns are out of vocabulary words. In this paper, a sequence learning method for transliterating named entities from Tamil to Hindi is proposed. Through this approach, accuracy obtained is encouraging. This transliteration system can be embedded with Tamil to Hindi machine translation system in future.

Keywords: Named entities, Transliteration, Phonetic, Alphabet, Sequence Labeling, Support Vector Machines.

1 Introduction

Named entity transliteration is the process of producing an equivalent target name for a given source name. In any machine translation system, large bilingual lexicons provide major coverage of words encountered in the text, a significant portion of tokens which are not covered by such lexicons is proper nouns [1]. Though bilingual lexicons are updated time to time, new named entities are still appearing frequently. Automatic transliteration is helpful in such cases. The aim of cross lingual information retrieval (CLIR) system is to retrieve documents in one language while the query is given in another language. The out of vocabulary word problem can be effectively handled with transliteration [2]. Recently, several methodologies have been developed for machine transliteration. Pushpak Bhattacharya et al., developed compositional transliteration system [3].They proposed the idea of compositionality of transliteration functionality in two different methodologies: serial and parallel. By composing serially two transliteration systems namely, X →Y and Y →Z, practical

V.V. Das, E. Ariwa, and S.B. Rahayu (Eds.): SPIT 2011, LNICST 62, pp. 262–264, 2012.

transliteration functionality between two languages X & Z which has no direct parallel data between them are provided and improving the quality of an existing X → Z transliteration system through a parallel compositional methodology. In this paper, focus is on Tamil to Hindi transliteration. The recent work by Vijaya M S et al., [4] on Machine Transliteration based on Sequence Labeling Approach for English to Tamil using Memory-based Learning has been adopted.

2 Issues in Tamil to Hindi Transliteration

Hindi is phonetically strong compared to Tamil. There are several issues that have to be considered while transliteration. Single alphabet in Tamil corresponds to more than one alphabet in Hindi. For example, க (ka)is mapped to क, ख, ग, घ(ka, kha, ga, gha). There are certain names in Tamil whose pronunciation varies in Hindi. For example, equivalent name for லட்சுமி (ladsumi) is लक्ष्मी(lakshmi).

3 Transliteration Using Support Vector Machines

3.1 Transliteration Framework

Support vector machines belong to supervised learning methods that analyze data and recognize patterns, which are used for classification and regression analysis [5]. The aim of the supervised learning is to provide output label for the given input based on the learned model. The input is given as a sequence and corresponding label is obtained as a sequence. Therefore this can be formulated as a sequence labeling approach. Table 1 shows the algorithm followed.

Table 1. Algorithm of our framework

```
X € {all possible transliterated units in Tamil}
Y € {all possible transliterated units in Hindi}
x→ Tamil word segmented as $(x_1,x_2...x_n)$
y→ Hindi word segmented as $(y_1,y_2...y_n)$
For i=1 to n
        $x_i \longrightarrow y_i$
        Each $x_i$ is mapped with its phonetically equivalent $y_i$
end
$y_i$ depends on,
{
            Source language unit $(x_i)$
            Adjacent units $(x_{i-2}, x_{i-1}, x_{i+1},x_{i+2})$ surrounding $x_i$
            Target language unit $(y_i)$
            Adjacent units $(y_{i-2}, y_{i-1}, y_{i+1},y_{i+2})$ surrounding $y_i$
  }
```

Corresponding to one unit, several output units are possible and hence, transliteration can be viewed as a multi-class classification problemWhile segmenting, twoapproaches have been carried out. In Approach I,Tamil alphabets are split along with its vowel. Approach II follows the same three steps as in the previous approach and also includes an additional step. One more level of splitting is done (i.e.) the splitting of consonants and vowels. In the previous approach, consonants and vowels are not split. More levels of splitting results in better accuracy. Accuracy obtained in Approach II is more compared to Approach I. After segmenting, corresponding Tamil and Hindi units are aligned. Alignment will be direct and easy if the number of units is same in Tamil and Hindi words. Problem will occur if there is mismatch in the number of units [4]. This mismatch is resolved either by inserting '$' symbol or by combining adjacent transliteration units of target side in such a way that phonetic structure is maintained. The other case will be to combine adjacent transliteration units when the number of units in Hindi word is more than that of the Tamil.

4 Experimental Results

Our model produces Hindi transliteration for Tamil words with an accuracy of 80-85% with Approach II and around 72% with Approach I. The character accuracy obtained in Approach II is 96.5% whereas for Approach I is 86.8%. The training data includes 30527 name and place names. The accuracy is affected because of the fact that, corresponding to a single character in Tamil, there are multiple transliterations possible in Hindi. Approach II gives better accuracy than Approach I. This methodology can be used for transliteration between any two languages.

References

1. Virga, P., Khudanpur, S.: Transliteration of Proper Names in Cross-Language Applications. In: 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2003)
2. Kumaran, A., Kellner, T.: A Generic Framework for Machine Transliteration. In: The 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2007)
3. Kumaran, A., Khapra, M.M., Bhattacharyya, P.: Compositional Machine Transliteration. ACM Transactions on Asian Language Information (2010)
4. Vijaya, M.S., Shivapratap, G., Dhanakshmi, V., Ajith, V.P., Soman, K.P.: Sequence labeling approach for English to Tamil Transliteration using Memory based Learning. In: International Conference on Natural Language Processing (2009)
5. Wikipedia, `http://en.wikipedia.org/wiki/Support_vector_machine`
6. Lin, W.-H., Chen, H.-H.: Backward Machine Transliteration by Learning Phonetic Similarity. In: The 6th Conference on Natural Language Learning, vol. 20 (2002)
7. TALP Research Center NLP group, `http://www.lsi.upc.edu/~nlp/SVMTool/`