



## A Review on Bengali to English Machine Transliteration System

Debashis Sen, Kamal Deep Garg  
Dept. of Computer Science  
Lovely Professional University,  
Phagwara, Punjab, INDIA

**Abstract:** Transliteration is the conversion of a text from one script to another. It is basically the conversion of one writing system to another. The characters from a source language are mapped to the characters of the target language. Designing a transliteration system is based on grapheme, phoneme, and correspondence and hybrid methods. In this paper, all these methods will be discussed. The idea is to develop a Bengali to English Machine Transliteration system using rule-based approach. A forward transliteration system for proper nouns, names of cities, states and rivers will be discussed in this paper.

**Keywords:** Transliteration, Siddham Script, Grapheme, Phoneme, Rule based.

### I. INTRODUCTION

The mapping of letters from between two similar scripts like English-French or between two dissimilar pair Bengali-English is the process commonly known as Machine Transliteration (MT). Any MT system aims to get zero aid from human. Problems in converting proper names and other technical named entities are faced by these systems. Transliteration of similar language pairs is easier than dissimilar language pairs. For instance English-Spanish pair, John-Juan. While a Bengali-English pair, বীমল-Vimal. An important aspect of any transliteration system is to be lossy in its approach. A utility keyboard is an example for application of these systems. A MT system follows some basic principles. It has to be easily readable, consistent results generated, approximation of the output, must be economic, should be partially reversible, uniform results and disputes between similar outputs must be resolved. Some of the guidelines that any transliteration system follows are absolute result, the calligraphy of the script should be adequate for the conversion, pronounceable, explicit and reversible.

### II. SIDDHAM SCRIPT

Bengali lingo was derived via the Siddham Script (Siddham) that is a descendent of the Brahmic kin of scripts. Brahmi is among the pre-historic writing systems. It was commonly used in the Indian subcontinent as well as in Central Asia for the period of the final centuries of 3<sup>rd</sup> century BCE to 5<sup>th</sup> century CE. Brahmi was an Abugida. Abugida is a figmental writing system where consonant-vowel combinations are inscribed as a lone unit. Every single unit is planted on a consonant letter and a vowel entry. Abugida includes the extensive Brahmic family of scripts of South and Southeast Asia. In 1837 the Brahmic Script was unravelled by James Prinsep. Late Brahmi is sometimes referred to the Gupta Script of the 5<sup>th</sup> Century. It is considered to be one of the most influential writing traditions. The Bengali lingo is inscribed in left to right direction. Uppercase letters are not present in the Bengali lingo. The Bengali script is specified by lots of concocts, apostrophes, down-strokes and other characteristics which suspend from a flat line. The Bengali script comprises of 29 consonants and 7 vowels.

### III. RELATED WORK

It gives a summary and brief insight of every work. The following literature review has been met out. Veerpal Kaur, Amandeep Kaur Sarao and Jagtar Singh (2014), has proposed a review on Hindi to English Transliteration system for proper nouns using hybrid approach. They proposed a model for backward machine transliteration from Hindi to English. In this system, the results are compared using two models i.e., direct mapping and rule based. For direct mapping they have obtained an accuracy of 47% and for rule based machine transliteration they have obtained an accuracy of 67%. [1] D. Bhalla, N. Joshi and I. Mathur (2013) developed a Rule Based Transliteration model for English to Punjabi. For syllabification, rules were generated. For named entities, likelihood was found out and those words that were not in the named entities were evaluated by means of their comparative occurrence. Vowel and consonant arrangement is taken as a lone syllable. Consonants ensued by a vowel, vowels ensued by two consonants and vowels bounded by two consonants are all considered as a lone syllable. They are next mapped into Punjabi. A sum of 1500 sentences with 15982 words was taken and an accurateness of 88.19% was obtained. [2] Chandranath Adak (2013) developed an Advanced Approach for Rule Based English to Bengali transliteration. Part of speech and hidden markov model are used as taggers. A fuzzy if then rule (FIFR) is used to select the lemma and consists of mainly universal Bengali words and their

equivalent English words. Both are joined into a united parallel multilingual corpus. Penn Treebank part of speech tagging is worn to split the sentence into its equivalent parts of speech. Words are categorised into simple n-grams. Reorganization of the words is completed past the words are tagged. FITR is used to pull out the lemma next to the processes information base in parallel alignment with the corpus. The transliteration is next carried out and united to get the equivalent Bengali decree. An accuracy of 81.5% F-score was obtained by this system. [3] Gupreet Singh Joshan and Jagroop Kaur (2011) developed a Punjabi to Hindi statistical machine transliteration. The system uses a baseline scheme to converts the generally happening words in Punjabi to the most happening words in Hindi which is listed as case 1. In case 2 a statistical model is generated wherein a parallel corpus was considered. This parallel corpus consists of the training information, and the development data. The training info and the development info have 8000 and 1125 entries. A third case wherein a person was endorsed to interpret the words with no exterior aid and his translated words was compared with the equivalent output. Using the baseline method, accuracy of 73.13% was obtained. Using the statistical machine transliteration model, an accuracy of 87.72% was obtained. [4] Kamal Deep and Vishal Goyal (2011) developed a hybrid based machine transliteration model from Punjabi to English. A collection of mapping for font or rules was related among the scripts. For English language, equivalent Roman script was used. For Punjabi lingo, equivalent Gurumukhi script was used. 1013 names of persons and city names, state names and river names amounting to 128 were used. For every character in the scripts, independent vowel, dependent vowel, consonant and special symbol mappings were considered. Diverse rules were generated for all the known mappings formed. When calculated using parameters of whole amount of words inputted and whole amount of words that are generated correctly, an accuracy of 95.00 % was obtained. For names of states, rivers and cities, 91.40 % was obtained. An overall of 93.22 % accuracy was obtained. [5] Naushad UrZaman and Arnab Zaheen (2009) developed a comprehensive roman to bangla transliteration scheme. This system implements a phonetic encoding model. This was used to generate interlaying code-strings which are responsible for the matching of pronunciations of the inputted data's and their corresponding results. They used lexicon enabled phonetic mapping and generating the code-strings by implementing double metaphone encoding model. If code strings are matched with the phonetic code strings to only a single word in the lexicon. The input is then generated to the desired Bengali word. If multiple words are matched, then it displays relevant bangla words corresponding with the input and lets the user select the correct output. If no match is found then the input is converted into bangla using direct mapping. 68% of the words only match leaving 32 % unmatched due to errors in direct mapping and infected words. [6] Malik, Besacier, Boitet, Bhattacharyya (2009) have developed Urdu-Hindi Transliteration system using hybrid approach which was a combination of statistical word language model (SWLM) using finite-state machine (FSM) techniques. In order to produce the efficient Hindi result, the feedback from the FSM has been refined with the SWLM. Different steps involved in system are pre-processing UHT-FSM Output, SWML and Word Map involving the Hindi script. This system has upgraded the accuracy of the transducer up to 79.1% from 50.7%. Results from this system indicate that the overall efficiency of these types of systems can be refined using a word language model. [7]

#### IV. TRANSLITERATION MODELS

Through the entry of novel learning and the virtuals of data all through the net, it has ended up being progressively more general to acknowledge outsider words into one's language. This ordinarily involves modifying the embraced word's unique articulation to take after the phonological guidelines of the target language, alongside adjustment of its orthographical. This transliteration of remote words is called transliteration. Transliteration is especially used to decipher fitting names and specialized terms from languages utilizing Roman letter sets into ones utilizing non-Roman letters in order, for example, from English to Korean, Japanese, or Chinese.

Four machine transliteration models have been proposed by several researchers:

A. *Grapheme based model ( $\Psi_G$ ):*  $\Psi_G$  is straight orthographical mapping from graphemes to graphemes. Graphemes are the most straightforward substance that may bring about a change in the significance in the written work framework. At the point when taken any two English words, for example, taste and dest, the whole importance of the word changes relying upon only two graphemes, "t" and 'd'. In the English letter set, there are twenty six letters in order. Every grapheme is encased with two precise sections to demonstrate their interesting significance like <t> and <d>. For Bengali letter sets that would be letters like <ক>, <ত> or <ী>. This model is in light of specifically mapping the images of the source language to its target dialect. Case in point to transliterate Mohan from English to Bengali, that would be <m>, <o>, <h>, <a> and <n> to <ম>, <ো> and <য> and <ন>.

B. *Phoneme based model (PBTM,  $\Psi_P$ ):* This model is in light of the elocution of the language instead of the letters. Phoneme is the littlest element in a language that may bring about a change in the meaning. The distinctive frameworks urbanized on the base of this counterfeit up are English to Korean MT framework,

Japanese to English MT framework, and so forth. The English to Korean MT framework at first changes over an English word into its explanation by method for a verbalization dictionary. It then fragments the English phonemes into bits of English phonemes; each part determines a Korean grapheme as denied off by means of high quality arrangement of laws. Finally, it mechanically changes over every part of English phonemes into its relating Korean graphemes by method for a thorough Markov sheet. It is taking into account the discourse resonances of the target and the source dialect and not the direct letter mapping. For example, the name transliteration of "bha" can be additionally composed as "bpha". That is the blend of /b/, /h/, /p/ and /a/. This idea was received in the correspondence based model. There are varieties of how phonemes are utilized. Short vowels (/æ/ in feline), schwa ('əmerikə in America), long vowels (/i:/ in week), diphthongs (/ai/ in sky), voiced consonants (/p/ in pot) and unvoiced constants (/t/ in tea). This model has issues in spreading slip.

*C. Hybrid approach ( $\Psi_H$ ):* This methodology consolidates the grapheme based and the phoneme based model methodologies amid the annexing of the source grapheme and the source phoneme utilizing linear interpolation. Linear interpolation is an approach to quantify the qualities that a capacity has more than two qualities. This approach takes the likelihood of the grapheme based model and the phoneme based model and afterward mixes them utilizing linear interpolation.

*D. Correspondence based model ( $\Psi_C$ ):* This methodology tries to utilize the first source grapheme furthermore the source phoneme in the transliteration. Case in point, the work "reek" has graphemes <r>, <e>, <e> and <k> and phonemes /w/, /e/, /e/ and /k/. Be that as it may, "reek" can be affirmed as "rik". For this situation, the grapheme <ee> is supplanted by a solitary phoneme /i/. Correspondingly in Bengali script in the event that we wish to change over Sari (which can likewise be proclaimed as Saree), it can be changed over as সারী or সরিঞ.

## V. PROPOSED MODEL

The methodology described in this proposal uses grapheme rule-based machine transliteration approach to convert words from Bengali to English. Direct mapping involves mapping the letters from the source language to the corresponding target language. Rules shall be generated for each corresponding case in which the vowel may be followed by a consonant, a vowel with a matra, a consonant with a matra, a consonant with a vowel etc.

### A. Direct mapping:

In this transliteration, the letters in Bengali and English are directly mapped using their Unicode. This is the basic step in any transliteration method. For each letters in the Bengali script the corresponding letter in English script is mapped. For example ক is converted to 'k', ব is converted as b, while কন is converted as 'kn', but it should be 'kan'. Similarly for বদল, it is converted as bdl, but the correct output should be badal. For this the table of independent vowels, dependent vowel mapping, consonant mapping and special character mapping are given in the tables below.

Example:

অ - a, ক - n, ন - n, ল - l

Table 1 Independent vowel mapping

| Bengali character | English alphabet | Bengali character | English alphabet | Bengali character | English alphabet |
|-------------------|------------------|-------------------|------------------|-------------------|------------------|
| অ                 | A                | উ                 | U                | ও                 | O                |
| আ                 | AA               | ঊ                 | UU               | ঔ                 | AU               |
| ই                 | I                | এ                 | E                |                   |                  |
| ঐ                 | II               | ঐ                 | AI               |                   |                  |

Table 2 Dependent vowel mapping

| Bengali character | English alphabet | Bengali character | English alphabet | Bengali character | English alphabet |
|-------------------|------------------|-------------------|------------------|-------------------|------------------|
| -                 | A                | ু                 | U                | ো                 | O                |
| া                 | AA               | ূ                 | UU               | ৌ                 | AU               |
| ি                 | I                | ে                 | E                |                   |                  |
| ী                 | II               | ৈ                 | AI               |                   |                  |

Table 3 Consonant mapping

| Bengali character | English alphabet | Bengali character | English alphabet | Bengali character | English alphabet |
|-------------------|------------------|-------------------|------------------|-------------------|------------------|
| ক                 | k                | খ                 | Ch               | দ                 | d                |
| খ                 | kh               | জ                 | J                | ঘ                 | dh               |
| গ                 | g                | ঝ                 | Jh               | ন                 | n                |
| ঘ                 | gh               | ঞ                 | Ny               | প                 | p                |

|    |    |    |    |   |    |
|----|----|----|----|---|----|
| ঙ  | ng | ত  | T  | ফ | ph |
| চ  | c  | থ  | Th | ব | b  |
| ভ  | bh | ম  | M  | য | y  |
| র  | r  | ল  | L  | স | s  |
| শ  | sh | য় | H  | ট | t  |
| ঠ  | th | ড  | D  | ঢ | dh |
| ণ  | n  | ঢ় | Rh | ষ | sh |
| ড় | r  |    |    |   |    |

Table 4 Table of special characters

| Bengali character | English alphabet |
|-------------------|------------------|
| ঈ                 | T                |
| ং                 | Am               |
| ঃ                 | H                |
| ঁ                 | N̄               |

**B. Rules used for transliteration**

Transliteration used in direct mapping gives wrong output. For this reason certain rules have to be generated for proper mapping of the letters to form the correct word. For instance when we write Kamal, in direct mapping its original input should be কমল. However the input comes out to be কামল if done by direct mapping. For this reason, rules have to be generated so that the word transliterated comes out in its original lingo form. Addition of the rules would aid in improving the accuracy of the system. Some of the rules generated are as follows

Rule 1: If any consonant is followed by another consonant and the latter consonant is combined with a matra I corresponding to 'ি'

Consonant + consonant + independent vowel I

হরি = হ+র+ি = Ha+Ra+i = Hari

সরি = স+র+ি = Sa+ra+i = Sari

Rule 2: If any consonant is followed by two other consonant.

Consonant + Consonant + Consonant

বমল = ব+ম+ল = Va+ma+la = Vamal

তনল = ত+ন+ল = Ta+na+la = Tanal

Rule 3: If a consonant comes with a hosonto.

Consonant + hosonto

ক + ং = k

ম + ং = m

Rule 4: If hosonto followed by a consonant and matra

Hosonto + Consonant + Matra

ং + ক + ী = Ki

ং + ক + া = Ka

Rule 5: If a consonant is followed by another consonant

Consonant + consonant

ড + ম = dam

ল + ল = lal

Rule 6: If a consonant is followed by a consonant and a matra

Consonant + consonant + matra

ক+চ+ি= kaci

দ+ল+ৌ=dalau

Rule 7: If a consonant is followed by a Chandra-bindu

Consonant + Chandra-bindu

ক+ঁ= kn

ম+ঁ= mn

Rule 8: If a consonant is followed by a anusvara

Consonant + anusvara

ক+ং=kam

ম+ং=mam

Rule 9: If a vowel is followed by a consonant

Vowel + consonant

অ+ক= ak

অ+ম=am

Rule 10: If a vowel is followed by two consonants

Vowel + consonant+ consonant

অ+ক+ল= akal

অ+ম+ল=amal

These are just some rules. In Bengali script there are total of 11 vowels and 40 consonants, so rules will be made for all the possible correct words that can be transliterated to English.

## VI. CONCLUSION

In grapheme rule-based approach, production of rules helps overcome the limitations of direct mapping technique. Production rules for dependent and independent vowels, consonants and special characters help in the proper transliteration of words from Bengali to English. Input of approximately 3000 will be taken for this system. The major drawback of this system is for conversion of words like পরজ to Vraj, Braj etc. Generally the conversion of পরজ is Varaj. So the problem is when to convert it to Varaj and when to Vraj.

## REFERENCES

- [1] A. k. S. a. J. S. Veerpal Kaur, "A Review on Hindi to English Transliteration System for Proper Nouns Using Hybrid approach," *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, vol. 3, no. 5, pp. 9-13, Sep. 2014.
- [2] N. J. I. M. Deepti Bhalla, "Rule Based Transliteration Scheme for English to Punjabi," *International Journal on Natural Language Computing*, vol. 2, no. 2, Apr. 2013.
- [3] C. Adak, "An Advanced Approach for Rule Based English To Bengali Machine Translation," *ACER*, pp. 01-09, 2013.
- [4] J. K. Gurpreet Singh Josan, "Punjabi to Hindi Statistical Machine Transliteration," *International Journal of Information Technology and Knowledge Management*, vol. 4, no. 2, pp. 459-463, Jul. 2011.
- [5] K. D. a. V. Goyal, "Development of a Punjabi to English transliteration system," *International Journal of Computer Science and Communication*, vol. 2, no. 2011, pp. 521-526, 2011.
- [6] N. U. a. A. Zaheen, "a comprehensive roman (english)-to-bangla transliteration scheme". "
- [7] B. B. a. B. Malik, "A Hybrid Model for Urdu Hindi Transliteration," *ACL-IJCNLP*, vol. 7, pp. 177-185, 2009.