AUTOMATIC TRANSLITERATION AMONG INDIC SCRIPTS USING CODE MAPPING FORMULA

Ahmad Hweishel AL-Farjat

Applied Science Department, AlBalqa Applied University, Jordan, Aqaba

Abstract:

This paper, discuss about developing an Automatic Transliterator which transliterates one Bramhi origin Indic script to another.

This methodology is used specifically to map a text from one writing system to another and does not map the sounds of one language to the best matching script of another language.

Here, we have discussed a method of Transliteration by finding out the patterns in Unicode Chart. It is mainly works for the following Indic Scripts: Bangla, Devanagari, Gurumukhi, Gujarati, Kannada, Malayalam, Oriya, Tamil, and Telugu. Each of these Indic Scripts can be interchangeably transliterated.

Keywords: Translation, Transliteration, Script, Arabic, Kannada

Introduction

As majority of population know more than one language, they understand the spoken or verbal communication, however when it comes to scripts or written communication, the number diminishes, thus a need for transliteration tools which can convert text written in one language script to another script arises.

Transliteration is mapping of pronunciation and articulation of words written in one script into another script. Transliteration should not be confused with translation, which involves a change in language while preserving meaning.

For example

Arabic: اللغة العربية هي إحدى أكثر اللغات انتشارًا في العالم

Translation: Arabic is one of the most widely spoken languages in the world

Arabic: اللغة العربية هي إحدى أكثر اللغات انتشارًا في العالم

Transliteration: allghh al'erbyh hy ehda akthr allghat antsharan fy al'ealm

History of Transliteration in Indian Languages: Ancient Indian inscriptions include a few bi-script documents, in which the text is given in the same language, written in two different scripts. Most of the instances come from the northwest and consists of short bi-script Bramhi-Kharoshthi inscriptions, the longer records include an 8th century Pattadakal Pillar inscription of the Chalukya king Kirttivarman II. The language is Sanskrit; the text is written both in the north Indian Siddhamatrika script and in the local southern proto-Telugu-Kannada script. [See Upinder Singh 2008]

Kannada is one among the Indian Classical Languages, spoken in Karnataka, which has its own script. Kannada speaking areas were administered by rulers who ruled multilingual states, like Mourya Empire, Vijayanagara Empire, Hydrabad Nizams, Mysore Princely State, and Ganga Dynasty. Kannada alphabet is developed from the Kadamba and Chalukya scripts, which are the descendants of Brahmi script which was used between the 5th and 7th centuries A.D. This script developed into the Old Kannada script, and was morphed into the Kannada and Telugu script in early 1500. Until the rivalry between the Yadava rulers of Devagiri in the upper Godavari region and the Yadava rulers of Hoysal in Karnataka became strong, the two languages, i.e. Marathi and Kannada were used generally in the area now known as Maharashtra. Several Old Marathi inscriptions are found in Kannada Script, while several old Kannada inscriptions are available in Devanagari. (See Madhav Deshpande 1993)

All Major Indic scripts are derived from Bramhi script, almost all the character mapping based on 128 Character slot is straight forward. However, there are cases when each scripts deviates from other scripts and such issues are discussed here.

Definition: From an information-theoretical point of view, systematic transliteration is a mapping from one system of writing into another, word by word, or ideally letter by letter. Transliteration attempts to use a one-to-one correspondence and be exact, so that an informed

reader should be able to reconstruct the original spelling of unknown transliterated words. Ideally, reverse transliteration is possible.

According to the Unicode Transliteration Guidelines (see http://cldr.unicode.org/index/cldrspec/transliteration-guidelines) "Transliteration is the general process of converting characters from one script to another, where the result is roughly phonetic for languages in the target script."

Transliteration from one Indic script to another is useful where one knows a particular language but not the script. Through Transliteration from one Indic script to another, he/she can read the script of that language in their respective native language script.

Transliteration schemes have to face the problem of letters present in one language and not in the other. Moreover, the large number of vowels in Indian scripts also adds to the complexity in transliteration.

Methodology: Automatic Transliteration mainly uses the pattern in the Unicode chart which is used to represent the character sets. In the Unicode Chart, Major Indic scripts codes ranges from 2304 to 3455 which was divided into 128 character-set. A character slot of 128 is assigned to each language script group. The characters or letters in one slot corresponds to the characters in the consecutive slots in other character slots.

E.g. the 128 character slot for Devanagari ranges from 2304 to 2431. Within this slot, the character, say 2309 which represents 'अ'('a') corresponds to the character in the next character set i.e. 2437 which is the Bengali 'অ'('a') of the next slot. The character slot for Bengali ranges from 2433 to 2554.

Thus, by recognizing the distance from the source character slot of 128 to the target language script, a uniform method can be formed in which each character in the character slot corresponds to the characters in the target language slot.

The Code Mapping Formula:

TargetCharCode = SourcecharCode + (TargetScriptBlock - SourceScriptBlock)*128

Where SourceCharCode = Character Unicode point of Source under consideration,

TargetCharCode = Character Unicode point of Target being computed,

SourceScriptBlock = 128 Unicode block of Source Script,

TargetScriptBlock = 128 Unicode block of Target Script.

In this method we are not considering the language scripts used in India which does not fall in Unicode Indic Script block range (2304 to 3455). i.e. Arabic (used for Urdu and Kashmiri), Ol Chiki (used for Santali), Meetei Mayek (used for Manipuri)

Challenges: The Unicode Chart for Devanagari uses Parivardhit Devanagari. The Parivardhit Devanagari Script is an enhanced set of symbols which has additional symbols introduced to take care of many of the symbols present in other Indian Scripts and not present in Devanagari.

"In 1966, the Government of India's Central Hindi Directorate produced an extended - Parivardhit - Devanagari to allow for other Indian languages to be represented in Devanagari (an approach which in principle would also allow all Indian languages to be represented in any of the other Indian scripts, such as Gujarati, Gurumukhi, Bengali, Oriya, Tamil, Telugu, Kannada and Malayalam)." (1995, Clews)

E.g. Devanagari ऐ (e) is used for ಎ/ എ/ ఎ in Kannada, Malayalam and Telugu respectively.

Even though these major Indic scripts are syllabic and derived from Bramhi script, each of them grew in timeline according to the need of language which uses the script; they got modified. Certain charcters missed out as the language using the script had no such phonetic property, included some characters for the need of some languages. So even though most of the characters mapped, there are issues when certain cases which is discussed here.

The Issue of Missing Characters in a script block: There are instances where there is no corresponding character available in Native Script. In such cases the IL2IL-Transliterators which follow the pattern of 128 Indic Script slot will render a junk character. E.g. 'ँ' (Chandrabindu) of Devanagari has no corresponding character in Kannada. In such a case, the Transliterator should be programmed in such a way that, whenever 'ँ' (Devanagari chandrabindu) is encountered, and if the target script is set to Kannada script will be mapped with 'ಂ' (Kannada Anusvāra) which is the most likely substitute mapping. However, care should be taken in such cases that it does not lead to over generalisation of the source script.

The Issue of Script Grammar: Script Grammars define the manner in which a script of a given language is to be written and especially in the case of Indian languages, the ways in which ligatures have to be constructed. Even though the Major Indian Scripts namely Bangla, Devanagari, Gujarati, Gurmukhi, Kannada, Malayalam, Odia, Tamil, Telugu are derived from Bramhi script as time passes they defined there own way of representing syllables, Dravidian language scripts wont use vowels in the medial or ending of the words (only vowel *matras* are used) where as some of Indo-Aryan languages uses vowels in the medial or final. The Classical Language Sanskrit which uses almost all the scripts. The script grammar once defined and mandated by the authoritative body of the state creates a standard which will serve as a guide for developers of fonts. Eventually it will also determine the manner in which all printed as well as digital material is prepared, thereby slowly bringing in one standardized manner of representing the given language.

The Issue of Script grammer arises when words which are borrowed from one language is rearrested in another, Even though In Kannada when *tatsam* (words borrowed from Sanskrit to other languages) words are used, most of the *Anunāsika* (Nasal) which are used in Sanskrit script, are replaced by *Anusvāra* (Nasal Diacritic). Native Kannada readers are used to *Anusvāra*. Since the Automatic Transliterator directly maps the Devanagari script to Kannada, *Anunāsika* in the Devanagari script will be retained in Kannada script also.

After transliterating Sanskrit in Devanagari to Kannada, retaining the *Anunāsika by* Proposed Automatic Transliteration schema.	Sanskrit written in Kannada Script following Kannada Script grammar where *Anunāsika's* are replaced by *Anusvāra*
ಮಾನಸೇನ ದುಃಖೇನ ಶರೀರಮುಪತಪ್ಯತೇ । ಅಯಃ**ಪಿಣ್ಡೇ**ನ ತಪ್ತೇನ **ಕುಮ್ಭ** ಸಂಸ್ಥಮಿವೋದ**ಕಮ್** (see Shrimanmahabharatha Vol-4, 2003)	ಮಾನಸೇನ ದುಃಖೇನ ಶರೀರಮುಪತಪ್ಯತೇ । ಅಯಃ**ಪಿಂಡೇ**ನ ತಪ್ತೇನ **ಕುಂಭ** ಸಂಸ್ಥಮಿವೋದ**ಕಂ**
Transliteration in Roman: mAnasEna duHkhEna sharIramupatapyatE । ayaHpiNDEna taptEna kumbha saMsthamivOdakam	

Translation in English: As the water in a pot gets heated up when a hot iron ball is immersed, so does the body ache when a person suffers from mental agony

- (Mahabharatha - Aranya Parva: Chapter-2, Verce-24)

Transliteration is done for easy reading of the native speaker, but to modify and fine-tune the source script even when target script supports the source script should be discouraged because the language being transliterated is evolved in its native script over a time with its culture and community. When the target script has supportive characters retaining the native script grammar in the target script is an easy and preferable choice

The Issue of Special Characters: In certain Indic scripts after 112th position of the script block in the Unicode chart, the characters are assigned to culture specific/language specific characters. These set of letters are termed as 'Other Letters' in Unicode chart. E.g. in Gurumukhi script, the character '੍ਓ' (Ikk Ōankār) is assigned the 117th place. '੍ਓ' (Ikk Ōankār) is a central tenet of Sikh religious philosophy. It is a symbol of the unity of God in Sikhism, and is found on all religious scriptures and places such as Gurdwaras. There is no corresponding character in other Bramhi-Indic scripts. In such cases, the source script should not be transliterated to keep up the integrity and chastity and significance of religious/cultural symbols of the source script, the source script should be retained. Same applies to Devanagari 'ॐ' or Tamil '' or Odia '୭'. A preprocessor should run beforehand of code mapping formula to keep such character codes in exception from getting transliterated.

Issue of Ambiguity in Source Script: Even though Tamil script (Unicode block 2944-3071) is bramhi script derived, it has a smaller set of characters compared to other Indic scripts. Same orthographic representation is read in many ways depending on context. Lets see an example of Tamil to Kannada Transliteation; Tamil 'க' (ka) will get mapped with Kannada 'ಕ' (ka). But Tamil 'க' (ka) can be used for 'ka' or 'ga' or 'ha') depending on the position-of-occurrence. Kannada has 'ಗ' (ga) and 'ಹ' (ha) separately. (See Manasa G, Rajesha N, Malini, 2011) To transliterate just the script 'க' without taking into consideration of position-of-occurrence in source will make transliteration less readable. A preprocessor should run

beforehand code mapping formula to re-arrange the character codes in Tamil script block according to the Tamil phonetic convention so that when code mapping formula runs it gets rightly (to best phonetically possible) transliterated.

Issue of Reversibility: The aim of transliteration is to be unique and therefore reversible. But this aim cannot always be achieved. If one alphabet has more characters than another one, a common practice is to use two characters to represent one. In case of Tamil the afore said common practice is also not useful because same character is representing more then one phonetic values depending on position of occurrence. So if other script gets transliterated to Tamil with such multi-valued character, the reverse transliteration won't give back the source script.

In Gurumukhi script used for the Punjabi Language uses ' ◌ੱ ' (Gurumukhi Tippy) to signify doubling of a consonant cluster. It needs to be preprocessed as doubled character cluster beforehand processed by code mapping formula. Similarly when a doubled consonant cluster from others scripts Gurumukhi script can be post processed for replacing any doubled consonant with a ' ◌ੱ ' (Gurumukhi Tippy), as an exception for the principle of following source scritpt grammar in target scritpt to make it easily readable.

In case of Gurumukhi Tippy and Phonetically Multivalued Characters of Tamil the principle of following source script grammar has to be relaxed as the characters in the scripts vary in phonetic property itself.

Issue of Multiple mapping: 'ৰ' (Assamese letter 'ra') maps with '॰' (Devanagari abbreviation sign), Tamil '൰' (digit 10), Odiya '୰'(isshar), Therefore, while mapping, multiple target scripts have to be considered and the source has to be transliterated according to the particular Target script. A preprocessor has to be designed in the automatic transliterating system which makes an internal mapping in the script block and then process through the aforesaid code mapping formula.

Enhancement: The transliterator system based on Code mapping formula can be enhanced for other non Indic Scripts easily. Just one among the script is manually mapped to the non-indic script and through a sub-routine it can be programmatically mapped for any other Indic

Script. For example Arabic to Devanagari, Meetei Mayek to Bangla and visa versa mapping can be used to transliterate for any Indic Script. Romanization of Indic script also can be thought of.

References:

Manasa G, Rajesha N, Malini, 2011, Automatic Transliteration from one Indic Script to another; with a 'Kannada' case study – In the *Proceedings of New Perspectives in Applied Linguistics*, Centre of Advanced Study in Linguistics, Annamalai University, pp 117 - 120

Upinder Singh, 2008, A History of Ancient and Early Medieval India: From the Stone Age to the 12th Century, Pearson Education India , pp 43

Madhav Deshpande, 1993, Sanskrit & Prakrit, sociolinguistic issues, Motilal Banarsidass Publication, pp 117

Rice, Edward. P (1921), "A History of Kanarese Literature", Oxford University Press, 1921: 14–15

Shrimanmahabharatha Vol-4 , 2003, Bharatha Darshna Prakashana, Bangalore. pp 1875

Clews, J. 1995. Asian Languages and Information Technology: A Summary of Issues for Newspaper Publishing. *Asia and Pacific Regional Seminar on Information Technology for Newspaper Publishing.* Madras: UNESCO

Lavanya, P., Kishore, P., Ganapathiraju, M. 2005. A simple approach for building transliteration editors for Indian languages. *Journal of Zhejiang University SCIENCE*. Hangzhou:Zhejiang University Press.

Surana, H., Singh. A. 2008. A More Discerning and Adaptable Multilingual Transliteration Mechanism for Indian Languages. In the *Proceedings of the Third International Joint Conference on Natural Language Processing*. Hyderabad.

Vijay, J. 2008. Transliteration Mapping in Intercultural Web Search. Extension of the paper presented at the Texas Linguistic Society conference in November 2006. (unpublished).

Acharya- *Transliteration Principles*. Multilingual Computing for Literacy and Education, SDL, IIT Madras.

Available at: http://acharya.iitm.ac.in/multi_sys/translit.php

Gogate M., 2003. A case of Roman Lipi for Indian Languages. *Language in India.* Vol. 3

Available at: http://www.languageinindia.com/march2003/roman.html