



International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS)

www.iasir.net

A Generic Transliteration tool for CLIA & MT Applications

Nishant Sinha, Atul Kumar and Vishal Mandpe
Computer Science Department
IMS Engineering College
Ghaziabad, India

Abstract: Transliteration has been a challenging problem in natural language processing specially in cross-lingual information access sub-domain. Transliteration from English as a source language becomes more tough due to non-phonetic property of English language. Different techniques are used to perform the task of efficient and correct transliteration. But till now rarely any achieve the goal. These techniques include the all four model based on source grapheme, source phoneme, target grapheme and target phoneme knowledge. Different approaches used for machine learning are maximum entropy model, decision-tree learning, memory-based learning, Genetic algorithm learning. Aim of the paper is to develop a tool that can be used as a sub-system of cross-lingual information access system. It takes a word or group of word as input and can generate the transliterated form of word so that output can be used for information access. This tool is platform independent, multi-pair transliteration system. It is something different from editor tools which also transliterate words but they can rarely give correct transliterated word as their first probable output. Editor tools only help in typing a document in our desired script and this process of editing is generally done by transliteration process because we have only one input medium which is English.

Keywords: Transliteration; GIST; Xlit; Google tool; Syllable Structure; Maximum Entropy Model

I. Introduction

Translation of proper names is generally recognized as a significant problem in many multi-lingual text and speech processing applications. Even when hand-crafted translation lexicons used for machine translation (MT) and cross-lingual information retrieval (CLIR) provide significant coverage of the words encountered in the text, a significant portion of the tokens not covered by the lexicon are proper names and domain-specific terminology. This lack of translations adversely affects performance. For CLIR applications in particular, proper names and technical terms are especially important, as they carry the most distinctive information in a query as corroborated by their relatively low document frequency. Finally, in interactive IR systems where users provide very short queries (e.g. 2-5 words), their importance grows even further.

Unlike specialized terminology, however, proper names are amenable to a speech-inspired translation approach. One tries, when writing foreign names in one's own language, to preserve the way it sounds. i.e. one uses an orthographic representation which, when "read aloud" by a speaker of one's language sounds as much like it would when spoken by a speaker of the foreign language — a process referred to as transliteration. Therefore, if a mechanism were available to render, say, an English name in its phonemic form, and another mechanism were available to convert this phonemic string into the orthography of, say, Hindi, then one would have a mechanism for transliterating English names using Hindi characters. The first step has been addressed extensively, for other obvious reasons, in the *automatic speech synthesis* literature.

The problem of transliteration of word aims to conserve the sound in target language as the sound of source language and transcript the word in target language. This process involves four steps:-[1]

1. Conversion of an English name into a phonemic representation
2. Translation of the English phoneme sequence into a sequence of generalized initials and finals or GIFs
3. Transformation of the GIF sequence into a sequence of Hindi sound units
4. Translation of the Hindi sound sequence to a character sequence.

II. Problem faced in transliteration from English to Hindi

Since Hindi uses a phonetic script (Devnagri), i.e., every character has one and only one pronunciation, transliterating from English to Hindi is essentially a process of determining the pronunciation of English words

and approximating them using the Devnagri script. Transliterating would have been an easy task if there were simple one-to-one rules for mapping sequences of characters in one language to another. But this is rarely the case. Many common sequences in English have different mappings based on the surrounding characters or the part-of-speech in which the word is used. In some cases, the differences are not even attributable to factors in the text, but are due to the etymology of the word or the arbitrary nature of pronunciation in English.

For example, c is pronounced differently in calendar, circus, and cheese; ea is pronounced differently in eat, threat, and heart; and many English multi-letter combinations are realized as a single sound in pronunciation (f, ff, ph, gh can be mapped to f).

The context usually plays a vital role in determining the pronunciation. The context can be the position, or the neighbourhood of the character sequence in terms of syllables, other sound units or characters.

With respect to position, if a word ends with c, then mostly it is pronounced as k (mosaic, disc, cardiac etc.). Similarly, if a word ends with e, then in most cases it remains silent without producing any sound (rise, promise, Jane, rose etc.). But context information is not adequate for disambiguating g in goat and ginger.

If we take a fixed number of characters as context, then the letter c followed by l, r or t is always pronounced as k. Also the phoneme ough is pronounced as af if its previous character is one of {c, d, n, r, t}. But, th is pronounced differently in though and thought even though it is followed by ough both cases.[2] Given these complexities a simple way of transliterating may be via lookup in a pronunciation dictionary. But, this will not work for new words and proper nouns. Also, any pronunciation dictionary would be correct only for the dialect it is created for. For example, an American English pronunciation dictionary would give awkward results for people who are used to the British accent. Thus, what we need is a system that can use a pronunciation dictionary or a corpus of transliterated examples and induce from this a set of rules. This can then be easily adapted for different dialects and accents provided we are able to find a sufficient number of training examples.

III. Test analysis of Different Transliteration Editor

We have tested four tool available for editing in hindi which uses the concept of transliteration for the purpose of editing in hindi. These are available as a web-based service. These four tools are:-

1. Xlit of CDAC-Mumbai
2. GIST of CDAC-Pune
3. Google Transliteration
4. Microsoft tool for hindi writing

Our test data are selected from different domain likes human names, building name, English name, city name, English words etc.

Test Results :-

English words

Roman script	देवनागरी लिपि			
	Xlit	GIST	Google	Microsoft
Abandon	अबंदों	अबॅन्डन	अबंदों	आबंदों
Abbreviate	अब्बरेवीते	अब्रीविएट	अब्ब्रेविअते	अब्ब्रेवियाते
Shrink	श्रृंक	श्रींगक	श्रिक	शृंक
Volcano	वोलकनो	वॉलकेनो	वोल्कानो	वोलकनों
Zooparasite	जूपरासाइट	जूपॅरसाइट	जूपरासिते	जूपरसीटे
Initial	इनीतील	इनीशिएल	इनितिअल	इनिश्यल
Injustice	इंजूसटिस	ईन्जस्टिस	इन्जुस्तिस	इन्जुस्तिके
Quotient	कोटेंट	क्वोशन्ट	उओतिएन्त	कुओटीएंट
Honour	होनौर	ओनर	होनौर	होनौर
Knight	कनाइट	कक्निघट	क्निघट	नाइट

Hindi Human names

Roman script	देवनागरी लिपि			
	Xlit	GIST	Google	Microsoft
John	जॉन	जॉन	जॉन	जॉन
Atul kumar	अतुल कुमार	अतुल कुमार	अतुल कुमार	अतुल कुमार
Raman	रमन	रामन	रमण	रमन
Rajan	राजन	राजन	राजन	राजन
Rajat	रजट	रजत	रजत	रजत
manu saxena	मन् सैक्सेना	मन् सक्सेना	मन् सक्सेना	मन् सक्सेना
prashant	प्रशांत खंडूरी	प्रशांत खंडूरी	प्रशांत खंडूरी	प्रशांत खनदूरी
megha jain	मेघा जैन	मेघा जैन	मेघा जैन	मेघा जैन
anil agrawal	अनिल अग्रवाल	अनिल अग्रवाल	अनिल अग्रवाल	अनिल अग्रवाल
amitabh bachchan	अमिताभ बच्चान	अमिताभ बच्चन	अमिताभ बच्चन	अमिताभ बच्चन
jaya bhaduri	जय भादूरी	जया भादूरी	जाया बहादूरी	जया भादूरी
kamal shrivastva	कमल श्रिवसत्व	कमल श्रीवास्तव	कमल श्रीवास्तव	कमाल श्रीवास्तव
ravi rituraj	रवि ऋतुराज	रवी ऋतुराज	रवि ऋतुराज	रवि ऋतुराज
Hans bhardwaj	हंस भार्दवाज	हंस भारदवाज	हंस भरदवाज	हंस भारदवाज
Shashi goswami	शशि गोस्वामी	शशि गोस्वामी	शशि गोस्वामी	शशि गोस्वामी
Roshan seth	रोशन सेठ	रोशन सेठ	रोशन सेठ	रोशन सेठ
Lata mangeshtakar	लट मंगेशकर	लता मांगेशाकर	लता मंगेशकर	लता मंगेशकर
Mohammad rafi	मोहम्मद राफी	मोहम्मद रफी	मोहम्मद रफी	मोहम्मद रफी
Sachin tendulkar	सचिन तेंदुलाकर	सचिन तेंदुलकर	सचिन तेंदुलकर	सचिन तेंदुलकर
Mahela	महेला जयवार्दने	महला जयवर्देने	महेला जयवार्देने	महेला जयवरदेने
Ram iqbal	राम इकबाल	राम इकबाल	राम इकबाल	राम इकबाल
Azizul haque	अजीजुल हकुए	अजीजुल हक	अजीजुल हकुए	अजीजुल हॉक
Vasistha narayan	वसीस्था नारायण	वासिस्था नारायण	वसिस्थानारायण	वसीस्था नारायण
Tenzin	तेंझिन प्रियदर्शी	तेनजिन प्रियादर्शी	तेनजिन प्रियदर्शी	टेंजीन प्रियदर्शी
amartya sen	अमर्त्य सेन	अमर्त्य सेन	अमर्त्य सेन	अमर्त्य सेन
Gautam Buddha	गौतम बुद्धा	गौतम बुद्ध	गौतम बुद्ध	गौतम बुद्ध
Aryabhata	अर्यभता	आर्यभट	अर्यभाता	आर्यभाता
Asvaghosa	अस्वाघोसा	अस्वघोस	अस्वघोसा	अस्वाघोसा
Devakinandan	देवकिनांडे खतरी	देवकीनंदन खत्री	देवकीनंदन खत्री	देवकिनन्दन खात्री
Bhartendu	भरतेंदु हरिसचन्द्र	भारतेन्दु हरिश्चंद्र	भारतेंदु हरिश्चंद्र	भारतेन्दु हरिश्चंद्र

English names

Roman script	देवनागरी लिपि			
	Xlit	GIST	Google	Microsoft
Christopher	क्रिसोफर मैनीग	क्रिसटोफर	चिस्तोफेर मंनिंग	क्रिस्टोफर
William	विलियम	विलियम	विल्लिं मच्छात्नी	विलियम
Barack obama	बरैक ओबामा	बरचक ओबामा	बरैक ओबामा	बराक ओबामा
John MacCain	जॉन मचैन	जॉन मचैन	जॉन मकैन	जॉन मककईन
Adam smith	अदाम स्मिथ	आदम स्मिथ	अदम स्मिथ	एडम स्मिथ
John rawl	जॉन रावल	जॉन रावल	जॉन रावल	जॉन रावल
Karl Marx	करल मारक्स	कार्ल मर	कार्ल मार्क्स	कार्ल मार्क्स
Immanuel Kant	इमानुएल कंट	इम्मनूएल कांत	इम्मानुएल कान्त	इम्मानुएल
Antonio Gramsci	अंतोऊनिय	एंटोनियो	अंतोनियो ग्राम्स्की	अंतोनिओ

William	विलियम	विलियम	विल्लिं शकेस्पारे	विलियम
Isaac newton	इसाक न्यूटन	आइज़क न्यूटन	इसाक नेव्त्तों	इसाक न्यूटन
Stephen hawking	स्टेफ़े हवकिंग	स्टीफन वी	स्टेफेन हव्किंग	स्टीफन हवकिंग

Building names

Roman script	देवनागरी लिपि			
	Xlit	GIST	Google	Microsoft
Redfort	रेडफोर्ट	रेदफोर्ट	रेडफोर्ट	रेडफोर्ट
Qutub minar	कुतुब मिनार	कुतुब मिनार	कुतुब मीनार	कुतुब मीनार
taz mahal	ताज महाल	तेज महल	ताज महल	ताज महल
rashtrapati	राष्ट्रपति भवन	राष्ट्रपति भवन	राष्ट्रपति भवन	राष्ट्रपति भवन
Parliament	परलियामेंट हाउस	पार्लियामेंट	पार्लिअमेंट हाउस	पार्लियामेंट
Madhur bhavan	मधुर भवन	मधुर भवन	मधुर भवन	मधुर भवन
Buland	बलैंड दरवाज़ा	बलंद दरवाज़ा	बलंद दरवाज़ा	बलंद दरवाज़ा
Golghar	गोलघर	गोलघर	गोलघर	गोलघर
Sanchi stupa	संघि स्तूप	सांची स्तूप	साँची स्तूप	सांची स्तूप
Lotus temple	लॉटस टेम्पल	लोटस टैपल	लोटस टेम्पले	लोटस टैम्पल

City names

Roman script	देवनागरी लिपि			
	Xlit	GIST	Google	Microsoft
Varanasi	वराणासी	वरानसी	वाराणसी	वाराणसी
Ara	अर	अरा	अर	अरा
Gwalior	गवालीओर	ग्वालियर	ग्वालियर	ग्वालियर
New delhi	न्यू डिलही	न्यू दिल्ली	न्यू डेल्ही	न्यू दिल्ली
Banglore	बंगलौर	बंगलोर	बंगलोरे	बंगलोरे
Chandigarh	चन्दीगरह	चंडीगढ़	चंडीगढ़	चंडीगढ़
Allahabad	अल्लहबद	इलाहाबाद	अल्लाहाबाद	इलाहाबाद
Lucknow	लकनौव	लकनाउ	लखनऊ	लखनऊ
Tiruchirappalli	तिरुचिरपपल्ली	तिरुछिरप्पल्लि	तिरुचिराप्पल्ली	तिरुचिरपपल्ली
thiruvananthapuram	ठिरुवानैटहपरम	थिरुवनन्थपरम	थिरुवानान्थापरम	थिरुवनन्थपरम

IV. Test Analysis

→Type of Word ↓ Transliterator	English Word		Hindi Human		English name		Building Name		City name	
	C	I	C	I	C	I	C	I	C	I
Xlit System	1	10	10	30	2	10	5	10	0	10
GIST system	8		23		3		6		6	
Google System	0		23		4		8		5	
Microsoft	1		21		6		10		7	

C - No of Correct output

I - No of Input

V. Comment on different System

Xlit:- This system is based on learning by Genetic Algorithm. So in process of pure learning it gives result based on rules derived by its system. It does not use the corpus directly to transliterated available word means It does not correctly transliterated the words from which this system learn as the dictionary word result by this system is poor which is about (1/10).

GIST:- This system gives good result for dictionary word. Perhaps it uses the learning corpus directly for transliterating dictionary words. It is also good for city names it means it has also a good parallel corpus of well-known named entities.

Google tool:- This system is based on pattern recognition from a large corpus of document available worldwide on internet . so its trained is good for all category of words except the English human name . it means there is less information is available on WWW by which it can learn to transliterate this type of words.

Microsoft tool:- Very less is known about this system but it gives good result. Dictionary words result of this system is poor. It is good in hindi names English names result is also poor . So it may be possible that it directly uses the fixed rules by which any hindi words is represented in English .

VI. Techniques which Can be explored

Syllabification Approach

Syllable Structure

1. Count of no. of syllables in a word is roughly/intuitively the no. of vocalic segments in a words.
2. Presence of a vowel is an obligatory element in the structure of a syllable. This vowel is called "nucleus".
3. Basic Configuration: (C)V(C).
4. Part of syllable preceding the nucleus is called the onset.
5. Elements coming after the nucleus are called the coda.
6. Nucleus and coda together are referred to as the rhyme.[3]

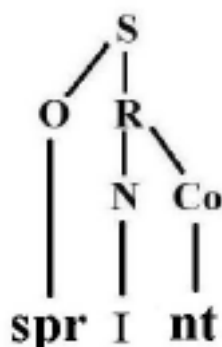


S = Syllable, O = Onset
R = Rhyme, N = Nucleus
Co = Coda

Ex- word

sprint

may



Maximum Entropy Model

In Bayesian probability, the principle of maximum entropy is a postulate which states that, subject to known constraints (called *testable information*), the probability distribution which best represents the current state of knowledge is the one with largest entropy.[4]

The principle was first expounded by E.T. Jaynes in two papers in 1957 where he emphasized a natural correspondence between statistical mechanics and information theory. In particular, Jaynes offered a new and very general rationale why the Gibbsian method of statistical mechanics works. He argued that the entropy of statistical mechanics, and the information entropy of information theory, are principally the same thing. Consequently, statistical mechanics should be seen just as a particular application of a general tool of logical inference and information theory.

In most practical cases, the testable information is given by a set of conserved quantities (average values of some moment functions), associated with the probability distribution in question. This is the way the maximum

entropy principle is most often used in statistical thermodynamics. Another possibility is to prescribe some symmetries of the probability distribution. An equivalence between the conserved quantities and corresponding symmetry groups implies the same level of equivalence for both these two ways of specifying the testable information in the maximum entropy method.

The maximum entropy principle is also needed to guarantee the uniqueness and consistency of probability assignments obtained by different methods, statistical mechanics and logical inference in particular. Strictly speaking, the trial distributions, which do not maximize the entropy, are actually not *probability* distributions.

The maximum entropy principle makes explicit our freedom in using different forms of prior information. As a special case, a uniform prior probability density (Laplace's principle of indifference) may be adopted. Thus, the maximum entropy principle is not just an *alternative* to the methods of inference of classical statistics, but it is an important conceptual generalization of those methods.

VII. Conclusion

Overall the task of transliteration is a tough task which require a learning process, then apply those learned concept to determine the new word's transliteration. It can give better result when we have a good learning technique. All the available tools are basically editor, they are not transliterator. They only help in writing contents in one's own language script only if he knows the manual transliteration of his content in English. No one can use these tools if he does not know English.

Here two approach discussed syllabification and maximum entropy model deals with the two aspect of transliteration process. Former deals with to reduce the context window of the letter i.e. in general if we consider context window size=3 then it means it consider the 7 letters for the context of a letter whether they all are in one syllable or not. If a letter in this context belong to other syllable of the word then it has no impact on the current considered letter's phoneme. Later is a learning method base on Bayesian probability

References

- [1] Virga and Khudanpur, 2003] Paola Virga and Sanjeev Khudanpur. Transliteration of Proper Names in Cross-Lingual Information Retrieval. In Proceedings of the Association for Computational Linguistics, 2003.
- [2] Automatic Derivation of Rules for Transliteration from English to Hindi: a Genetic Algorithm Approach, Alekha Mishra, Ananthakrishnan R, Sasikumar M CDAC-Mumbai.
- [3] Shankar Ananthakrishnan ,Statistical Syllabification of English Phoneme Sequences using Supervised and Unsupervised Algorithms.
- [4] K. Knight and J. Graehl. Machine Transliteration. In *Computational Linguistics*, pages 24(4):599–612, Dec. 1998.
- [5] Nasreen Abdul Jaleel and Leah S. Larkey. Statistical Transliteration for English-Arabic Cross Language Information Retrieval. In *Conference on Information and Knowledge Management*, pages 139–146, 2003.
- [6] Lee-Feng Chien Long Jiang, Ming Zhou and Chen Niu. Named Entity Translation with Web Mining and Transliteration. In *International Joint Conference on Artificial Intelligence (IJCAL- 07)*, pages 1629–1634, 2007.
- [7] Slaven Bilac and Hozumi Tanaka. Direct Combination of Spelling and Pronunciation Information for Robust Back-transliteration. In *Conferences on Computational Linguistics and Intelligent Text Processing*, pages 413–424, 2005.