Hybrid Approach for Punjabi to English Transliteration System

Kamal Deep
Dept of Computer Science
Punjabi University
Patiala, India

Dr.Vishal Goyal
Assistant Professor
Dept of Computer Science
Punjabi University, Patiala, India

ABSTRACT

Language transliteration is one of the important area in natural language processing. Accurate transliteration of named entities plays an important role in the performance of machine translation and cross-language information retrieval processes. The transliteration model must be design in such a way that the phonetic structure of words should be preserve as closely as possible. We have developed hybrid (statistical +rules) approach based transliteration system of person names; from a person name written in Punjabi (Gurumukhi Script), the system produces its English (Roman Script) transliteration. Experiments have shown that the performance is sufficiently high. The overall accuracy of system comes out to be 95.23%.

Keywords

Transliteration, Mapping, Dictionary

1. INTRODUCTION

Machine transliteration is very important for research in natural language processing (NLP), such as machine translation (MT), cross-language information retrieval (CLIR), question answering (QA), and bilingual lexicon construction. Transliteration system converts an input string to a string in target alphabet, usually based on the phonetics of the original word. Transliterating a word from the language of its origin to a foreign language is called Forward Transliteration, while transliterating a loan word written in a foreign language back to the language of its origin is called Backward Transliteration. In this paper we will discuss the Punjabi to English Machine transliteration system which is forward transliteration system. We will use letter to letter mapping as baseline and try to find out the improvements by statistical methods.

The remainder of this paper is organized as follows. In section 2, we have described the related work. Gurmukhi & Roman script is discussed in section 3. Design and Implementation of our system is discussed in sections 4. Evaluation and Results is discussed in section 5. Finally, we have concluded it in section 6.

2. RELATED WORK

The topic of Machine transliteration has been studied extensively for several different language pairs, and many techniques have been proposed. In Grapheme based approaches (Ψ_{G}), transliteration is viewed as a process of mapping a grapheme sequence from a source language to a target language ignoring the phoneme-level processes. In contrast, in phoneme-based approaches (Ψ_P), the transliteration key is pronunciation or the source phoneme rather than Spelling or the source grapheme. This approach is basically source grapheme-to-source phoneme transformation and source phoneme-to-target grapheme transformation. In hybrid approaches (Ψ_H), it simply combines the grapheme-based transliteration probability (Pr (Ψ_G)) and the phoneme-based transliteration probability (Pr (Ψ_P)) using linear interpolation.

Vijaya, VP, Shivapratap and KP CEN [1] has developed English to Tamil Transliteration system and named it WEKA. It is a Rule based system and is used the j48 decision tree classifier of WEKA for classification purposes. The transliteration process consisted of four phases: Preprocessing phase, feature extraction, training and transliteration phase .The accuracy of this system has been tested with 1000 English names that were out of corpus. The transliteration model produced an exact transliteration in Tamil from English words with an accuracy of 84.82%. Haque et. al [2] have developed English to Hindi Transliteration system based on the phrase-based statistical method (PB-SMT). PB-SMT model has been used for transliteration by translating characters rather than words as in character-level translation systems. An improvement of 43.44% and 26.42% has been reported respectively for standard and larger datasets. Jia, Zhu, and Yu [3] have developed Noisy Channel Model for Grapheme-based Machine Transliteration. They have experimented this model on English-Chinese. Moses, a phrase-based Statistical Machine Translation tool, has been employed for the implementation of this system. Both English-Chinese forward transliteration and back transliteration has been studied. The process has been divided into four steps: language model building, transliteration model training, weight tuning, and decoding. The orders with the best performance in all metrics for forward and back transliteration are 2 and 5.Chinnakotla, Damani, Satoskar[4] has developed Transliteration systems for Resource Scarce Languages. They have developed rule based systems for Hindi to English, English to Hindi, and Persian to English transliteration tasks. They used CSM (Character Sequence Modeling) on the source side for word origin identification, a manually generated non-probabilistic character mapping rule base for generating transliteration candidates, and then again used the CSM on the target side for ranking the generated candidates. The overall efficiency by using CRF (Conditional Random Field) approach of English to Hindi is 67.0%, Hindi to English is 70.7% and Persian to English is 48.0% .Chinnakotla and Damani[5] has developed Transliteration systems for English to Hindi, English to Tamil and English to Kannada. They used a Phrase-Based

Statistical Machine Translation approach for Transliteration where the words are replaced by characters and sentences by words. They have used news 2009 data, as development and test set. The overall accuracy of English to Hindi is 49.0%, English to Tamil is 41.0% and English to Kannada is 36.0%. The use of monolingual Hindi corpus in the non-standard run, the transliteration accuracy has been improved by 22.5% when compared to standard run. Lehal and Singh [6] have developed Shahmukhi to Gurmukhi Transliteration System based on Corpus approach. In this system, first of all script mappings has been done in which mapping of Simple Consonants, Aspirated Consonants (AC), Vowels, other Diacritical Marks or Symbols are done. This system has been virtually divided into two phases. The first phase performs pre-processing and rule-based transliteration tasks and the second phase performs the task of post-processing. The overall accuracy of system has been reported to be 91.37%. Malik [7] has developed Punjabi Machine Transliteration (PMT) system which is rule-based. PMT has been used for the Shahmukhi to Gurmukhi Transliteration System. PMT has preserved the phonetics of transliterated word and the meaning of transliterated word. The primary limitation of this system is that this system works only on input data which has been manually edited for missing vowels or diacritical marks (the basic ambiguity of written Arabic script) which practically has limited use. The accuracy of system has been reported to 98.95%. Verma[8] has developed Gurmukhi to Roman Transliteration System and named it GTrans. He has surveyed existing Roman-Indic script transliteration techniques and finally a transliteration scheme based on ISO: 15919 transliteration and ALA-LC has been developed. It is a rule based system. He has also done reverse transliteration from Gurumukhi to Roman. The overall accuracy of system has been reported to be 98.43%. UzZaman, Zaheenand, Khan [9] has developed Roman (English) to bangla transliteration system. Two mappings, one is direct phonetic mapping and second location enabled phonetic mapping, has been used. The user is provided with multiple options of letter-groups in the source script (which, in this case, is Roman) to represent one letter in the goal script (Bangla), (many-to-one mapping scheme). This scheme can be used in applications such as cross language information query and retrieval. Knight, Graehl [10] has developed English-Japanese Transliteration system. This system is a phoneme based as they converted English word to English sounds and then into Japanese sound. Japanese frequently imports vocabulary from other languages, primarily (but not exclusively) from English. It has a special phonetic alphabet called Katakana, which is used primarily (but not exclusively) to write down foreign names and loanwords. When they used OCR, accuracy only drops from 64% to 52%.

Sato [11] has developed a English to Japanese web-based transliteration system of person names i.e. from a person name written in English, the system produce its Japanese (Katakana) transliteration. This system is based on the hybrid approach. Tsumugi Finder has been worked as a transliterator in the system but it does not have any rules to produce transliterations; it just searches and extracts the transliteration pairs on the Web. Tsumugi also worked in reverse direction, i.e., back-transliteration from Japanese (Katakana) to English. Experiments have shown that the performance is sufficiently high i.e.89.4% of English person names, and the system produced 98.5% acceptable transliterations. Hong, Kim, Lee and Chang [12] have developed English-Korean Name Transliteration system, using the Hybrid Approach. In the transliteration process, first, a phrase-base SMT model with some factored translation features has been used. Second, they have expanded the base system by applying web-based *n*-best re-ranking of the results. Third, they have applied a pronouncing dictionary-based method to the base system which utilizes the pronunciation symbols which is motivated by linguistic knowledge. Finally, phonics based method is applied which has been originally designed for teaching speakers of English to read and write that language. The experimental results of using three n-best re-ranking techniques have showed that the web-based re-ranking is proved to be a useful method .Their standard run and best standard run has accuracy of 45.1% & 78.5%. Ali and Ijaz [13] have developed English to Urdu Transliteration System based on the mapping rules. The whole process has three steps. In the first step, the mapping rules that have been used to generate Urdu text from English transcription. English text is converted to Urdu using both English pronunciation and mapping rules. In Second step, Urdu syllabification has been applied on English transcription. Consonant and Vowels have been combined to make syllable and breaking up a word into syllables is known as syllabification. To improve system's accuracy, they have applied the Urduization Rules in third step. Overall system's accuracy is 95.9%. Hoon Oh, and Choi [14] have developed English-Korean Transliteration system using the hybrid approach, because it has used both phonetic information such as phoneme and its context and orthography. This method has been composed of two phases i.e. alignment and transliteration. First, an English pronunciation unit and its corresponding phoneme have been aligned phonetically. Second, English words have been transliterated into Korean words through several steps. Finally, Korean transliterated words have been generated using conversion rules. Evaluation has been performed through Word Accuracy (WA) and Character Accuracy (CA). This system has reported accuracy of 90.82% for WA and 56% for CA. Yaser, Knight [15] has developed Arabic to English Transliteration system based on the sound & spelling mapping using finite state machine. They have combined the phonetic based model & spelling based model into the single transliteration model. For testing they have used the development data set & blind data set. The overall accuracy with development data set has been reported to be 53.66% & with blind data set it showed 61% accuracy. The reason of high accuracy with blind data set was that blind set is mostly of highly frequent, prominent politicians where as development set also contain names of writers and less common political figure.

3. GURMUKHI & ROMAN SCRIPT

In this section we will discuss about Gurmukhi and Roman Script.

3.1 Gurmukhi Script

Punjabi Language is written in Gurmukhi Script. The Gurmukhi script was derived from the Sharada script and standardized by Guru Angad Dev in the 16th century. It was designed to write the Punjabi language. The meaning of Gurmukhi is "from the mouth of the Guru". The Gurmukhi (or Punjabi) alphabet contains thirty-five distinct letters. These are:

ੳ	ਅ	ੲ
Ura	Era	Iri

The first three letters are unique because they form the basis for vowels and are not consonants. Apart from Era, these characters are never used on their own.

ਸ	ਹ	ਕ	ਖ	ਗ	ਘ
Sussa	Haha	Kukka	khukha	Gugga	ghugga
ਙ	ਚ	ਛ	ਜ	ਝ	ਞ
ungga	chucha	chuchha	Jujja	jhujja	yanza
ਟ	ਠ	ਡ	ਢ	ਣ	ਤ
tainka	thutha	dudda	dhudda	nahnha	Tutta
ਥ	ਦ	ਧ	ਨ	ਪ	ਫ
thutha	duda	dhuda	Nunna	puppa	phupha
ਬ	ਭ	ਮ	ਯ	ਰ	ਲ
bubba	bhubba	mumma	Yaiyya	rara	Lulla
ਵ	ੜ				
vava	rahrha				

In addition to these, there are six consonants created by placing a dot (bindi) at the foot (pair) of the consonant:

ਸ਼	ਖ਼	ਗ਼	ਜ਼	ਫ਼	ਲ਼
Shusha pair bindi	Khukha pair bindi	Gugga pair bindi	Zuzza pair bindi	Fuffa pair bindi	Lulla pair bindi

In addition to this, there are nine dependent vowel signs used to create ten independent vowels with three bearer characters: Ura ੳ[ʊ], Aira ਅ [ə] and Iri ੲ[ɪ].

ੳ	ਅ	ੲ
Ura	Era	Iri

3.2 Roman Script

English Language is written in Roman script**.** English is a West Germanic language that arose in the Anglo-Saxon kingdoms of England. It is one of six official languages of the United Nations. India is one of the countries where English is spoken as a second language. There are 26 letters in English. Out of which 21 are consonants and 5 are Vowels. Vowels are:-

A	E	I	O	U

Consonants are:-

B	C	D	F	G	H	J	K	L	M	N
P	Q	R	S	T	V	W	X	Y	Z	

4. DESIGN AND IMPLEMENTATION

The system architecture, given below in Figure 1, consists of various stages through which source language text has to be passed to be converted into target language.

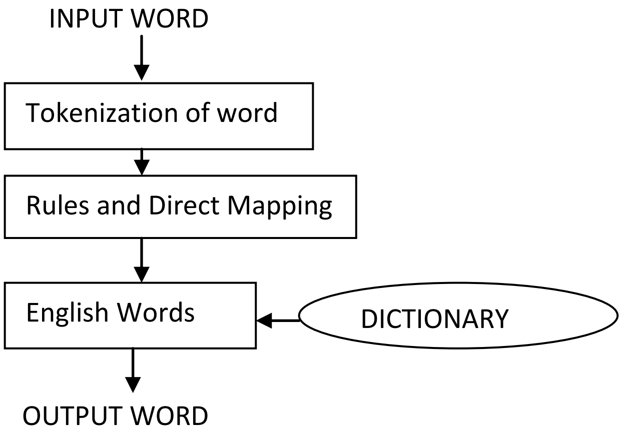

Fig 1: Architecture of Transliteration system

1. Tokenization:-This is the first layer of our system, which gets Punjabi word as a input and break it into the individual character. For example ਕਬੀਰ will be tokenized as ਕ, ਬ, ◌ੀ, ਰ.

2. Rules and Direct Mapping: - On the tokenized word, rules are applied. We have developed the 41 rules for Punjabi to English Transliteration. Rules in our system include constraints which specify the context in which they are applicable like Start of a Word (S), ending of a Word (E), After Vowel (AV), and After Consonant (AC) etc. If none of rule is applicable on the word then direct mapping is applied. Reason to develop rules is that if we used direct character mapping, then accuracy of system is very low. To improve that accuracy we have developed different rules. Combination of different mapping options for each character in inputting Punjabi words results in different output words. For ਕਬੀਰ we get (kabir,kabeer) as output.

Direct mapping of vowels and consonant is shown in Table 1, Table2,Table3,Table4.

3. Compression with Dictionary:-we have created a dictionary that contains the spellings of names that are commonly used in real life. For writing ਕਬੀਰ in English we write as kabir .Output of names from previous step is compared with this dictionary and system will show only that spelling of those names that is comman. If the no match found with dictionary then system will display output of second step as final output.

Table1: Independent Vowels Mapping

Gurumukhi char	English Letter	Gurumukhi char	English Letter	Gurumukhi char	English Letter
ਅ [a]	a	ਉ[u]	u	ਓ[ō]	o
ਆ[ā]	a	ਊ[ū]	u	ਔ[au]	au
ਇ[i]	i,ya	ਏ[ē]	a,e		
ਈ[ī]	i	ਐ[ai]	ai		

Table2: Dependent Vowels Mapping

Gurumukhi char	English Letter	Gurumukhi char	English Letter	Gurumukhi char	English Letter
◌ਾ[ā]	a	◌ੁ[u]	u	◌ੈ[ai]	ai,ay
ਿ◌[i]	i	◌ੂ [ū]	u,oo	◌ੋ[ō]	o
◌ੀ[ī]	i,ee	◌ੇ[ē]	e	◌ੌ[au]	o,au

Table 3: Consonant Mapping

Gurumukhi char	English Letter	Gurumukhi char	English Letter	Gurumukhi char	English Letter
ਕ[ka]	k	ਢ[ḍha]	Dh	ਰ[ra]	r
ਖ[kha]	kh	ਣ[ṇa]	N	ਲ[la]	l
ਗ[ga]	g	ਤ[ta]	T	ਵ[va]	v,w
ਘ[gha]	gh	ਥ[tha]	Th	ੜ[ṛa]	rh,r
ਙ[ṅa]	ng	ਦ[da]	D	ਸ[sa]	s
ਚ[ca]	ch	ਧ[dha]	Dh	ਹ[ha]	h
ਛ[cha]	chh	ਨ[na]	N	ਸ਼[sha]	sh
ਜ[ja]	j	ਪ[pa]	P	ਖ਼[k̲h̲a]	khh
ਝ[jha]	jh	ਫ[pha]	f,ph	ਗ਼[ga]	ghh
ਞ[ña]	yan	ਬ[ba]	B	ਜ਼[za]	z
ਟ[ṭ]	t	ਭ[bha]	Bh	ਫ਼[fa]	f
ਠ[ṭha]	th	ਮ[ma]	M	ਲ਼[ḷa]	lla
ਡ[ḍa]	d	ਯ[ya]	Y		

Table 4: Mapping of Special Symbols

◌ਂ[ṃ]	n
◌ੰ[ṃ]	un,an
◌ੱ[Doubles the following Character]	Doubles the following Character

5. EVALUATION AND RESULTS

In this section, we will discuss the accuracy of our system.

5.1 Evaluation Data

We have divided the data set into two parts. One is Training data set and second is Test Data set. Training data set consisted of names, using these names we have made the rules for the transliteration from Punjabi to English. And in Test Data set we have used the original data where it will be implemented. Our system is accurate for the Punjabi words but not for the foreign words. For evaluating the system we took names from the different domains like Person names, City names, State names, River names.

Table 5: Statistics of Dataset

Set	No of Examples
Training	1116
Test	2134

5.2 Evaluation Metrics

Transliteration accuracy rate is used for evaluation to capture the performance at word level. Accuracy Rate is the percentage of correct transliteration from the total generated transliterations by the system.

$$\text{Accuracy Rate} = \frac{\text{Number of Correct Transliteration}}{\text{Total no of Generated Transliteration}} * 100\%$$

5.3 Result

The result of test case is discussed given below. The overall accuracy of our system is 95.23%.

Table 6: Transliteration Accuracy Rate

Test Case	TAR
Using Rules and Direct Mappings**(Case 1)**	90.0%
Using Rules, Direct mapping and Compare with Dictionary**(Case 2)**	95.23%

This following figure gives us a graphical view of the accuracy.

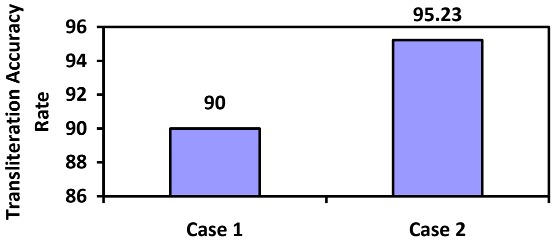

Figure 2

5.4 Error Analysis

The overall performance accuracy test of the system is quite good. There are several reasons for the errors in the output.

- **Multiple Transliterations**: Sometimes when a name is pronounced in Punjabi it correspond to many English words, so their system fails to guess which one is the best for that particular transliteration.
- **Wrong Input of Words:** Some time user does not enter correct data to the system due to which output is also not correct. For example ਬਿਲ੍ਲੂ ਸਿੰਘ as here halant is used as such but we know it is used to write half letter.
- **Character Gap**: The number of characters in, both English and Punjabi, character sets varies in both the language that makes the transliteration process difficult. The numbers of vowels are 5 and 20 and numbers of consonants are 21 and 41, in both English and Punjabi, respectively as explained earlier. So, there is character gap in both the languages that leads to problems in transliteration process. For Example, for character 'ਙ' in Punjabi there is no corresponding character in English.
- **One-to-Multi mapping Problem**: In this problem, single character in one script transform to multiple characters in another script. The Multi-mapping Problem is associated with following characters as shown in Table 3. For example, character 'ਫ' in Punjabi language can be transliterated into two characters in English 'ph' or 'f'. Some algorithm is required to select the appropriate character at different situations.

The following names are transliterated by our system.

Punjabi Word	Rule and Direct Mapping	Statistical approach
ਪ੍ਰਦੀਪ	pradeep, pradip	pradeep
ਸਤਿਆ	satia,satya	satya
ਗੰਗਾ	Ganga	Ganga
ਪੰਜਾਬ	punjab,panjab	punjab
ਏਕਲਵਯ	eklavya,aklavya	eklavya
ਵਿਸ਼ਾਲ	vishal,wishal	vishal
ਉੱਤਮ	Uttam	Uttam
ਪਟਿਆਲਾ	patiala,patyala	patiala
ਅੰਬਾਲਾ	Ambala	Ambala

Our system is efficient for the words that have the Punjabi origin not the foreign words that are written in Punjabi.

Punjabi Name	English Name	Our System
ਲਵ	Love	Lav ,law
ਮੈਂਕੀ	Manky	Mainki ,maynki
ਮੈਰੀ	Merry	Mairi,mayri
ਲੱਕੀ	Lucky	lakki
ਬਾਬੀ	Bobby	babi
ਜਿੰਨੀ	Jinny	jinni
ਹਕੀਕਤ	Haqiqat	Hakikat,haeekikat
ਪ੍ਰਿੰਸ	Prince	prins
ਟਵਿੰਕਲ	Twinkle	tawinkal

6. CONCLUSION

In this paper we have addressed the problem of transliterating Punjabi to English language using statistical rule based approach. Punjabi to English transliteration system is very beneficial for removing the language and scriptural barrier. The system is giving promising results and this can be further used by the researchers working on Punjabi and English Natural Language Processing tasks. As we know that in Punjab area most of the government departments use Punjabi language to store their data, so this transliteration system will help them a lot to transliterate Punjabi to English on a click of a button.

8. REFERENCES

[1] Vijaya ,VP, Shivapratap and KP CEN(2009) "*English to Tamil Transliteration using WEKA system"* International Journal of Recent Trends in Engineering, May 2009, Vol. 1, No. 1, pages: 498-500.

[2] Haque, Dandapat, Srivastava, Naskar and Way (2009) "*English—Hindi Transliteration Using Context-Informed PB-SMT: the DCU System for NEWS 2009"* Proceedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, pages 104–107,Suntec, Singapore, 7 August 2009. ACL and AFNLP.

[3] Jia, Zhu, and Yu(2009), "*Noisy Channel Model for Grapheme-based Machine Transliteration*", Proceedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, pages 88–91.

[4] Chinnakotla, Damani, Satoskar(2009) *"Transliteration for Resource Scarce Language*" ,ACM Transactions on Asian Language Information Processing, Vol. V, No. N.

[5] Chinnakotla and Damani(2009) "*English-Hindi, English-Tamil and English-Kannada Transliteration Tasks*".

[6] Lehal and Singh (2008) *"Shahmukhi to Gurmukhi Transliteration System: A Corpus based Approach"* proceeding of Advanced Centre for Technical Development of Punjabi Language, Literature & Culture,Punjabi University, Patiala 147 002, Punjab, India, pages:151-162.

[7] Malik(2006) *"Punjabi Machine Transliteration System":* In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL (2006) pages:1137-1144.

[8] Verma(2006) *" A Roman-Gurmukhi Transliteration system"* proceeding of the Department of Computer Science, Punjabi University, Patiala, 2006.

[9] UzZaman , Zaheenand ,Khan(2009) "*A Comprehensive Roman (English)-To-Bangla Transliteration Scheme"* A Comprehensive Roman (English) to Bangla Transliteration Scheme, Proc. International Conference on Computer Processing on Bangla (ICCPB-2006), 17 February, 2006, Dhaka, Bangladesh.

[10] Knight, Graehl (2005) *"English-Japanese Transliteration system"* Computational Linguistics, Volume 24,Number 4, pages:599-612.

[11]Sato (2009) " *Web-Based Transliteration of Person Names"* IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology – Workshops., pages:273-278

[12] Hong, Kim, Lee and Chang(2009) *"A Hybrid Approach to English-Korean Name Transliteration"* , Procedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, pages 108–111,Suntec, Singapore, 7 August 2009 ACL and AFNLP.

[13] Ali and Ijaz(2009), "*English to Urdu Transliteration System",* Proceedings of the Conference on Language & Technology 2009, pages: 15-23.

[14] Hoon Oh, and Key-Sun Choi (2002) *" An English-Korean transliteration model"* using pronunciation and contextual rules. In Proc. of the 19th International Conference on Computational Linguistics (COLING 2002), pages: 393–399.

[15] Yaser ,Knight(2002), *"Machine Transliteration of names in Arabic text",* Machine transliteration of names in Arabic text In Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, Philadelphia, PA,pages: 1-13

[16]"Transliteration" Internet Source:- *http://en.wikipedia.org/wiki/transliteration acessed on jan,2011*