TRANSLITERATION BETWEEN ENGLISH AND OTHER INDIAN LANGUAGES: A MACHINE LEARNING BASED APPROACH

A Synopsis of the proposed thesis to be submitted for the degree of
DOCTOR OF PHILOSOPHY
in
COMPUTER SCIENCE

Submitted by

Radha Mogla

Under the supervision of

Dr. C.Vasantha Lakshmi
Supervisor
Associate Professor
DEPT. OF PHYSICS & COMPUTER SCIENCE
FACULTY OF SCIENCE , DEI

Prof. Niladri Chatterjee
Co-supervisor
DEPT. OF MATHEMATICS
IIT DELHI

FORWARDED BY

Prof. G.S. Tyagi
HEAD
DEPT. OF PHYSICS & COMPUTER SC.

Prof. Ravindra Kumar
DEAN
FACULTY OF SCIENCE

DEPARTMENT OF PHYSICS AND COMPUTER SCIENCE
FACULTY OF SCIENCE
DAYALBAGH EDUCATIONAL INSTITUTE
(Deemed University)
DAYALBAGH, AGRA (UP) – 282005
APRIL 2016

CONTENTS

1.0. INTRODUCTION

In today’s time, global interactions are increasing day by day and communications between different nationals are done in different languages as well. No person knows all the languages and scripts. Although English is a global language, not everyone understands it and not every document is available in English. To overcome this barrier of language, translation is one very important tool.

The process of converting a text written in one language to another without changing its meaning is known as translation. Thus, a word in Roman script (English language) “School” when translated to Devnagari script (Hindi) becomes “विद्यालय” read as “Vidyalaya” and the same when translated to Telugu, becomes పా ఠశా ల(“Pathshala”).

Machine translation system is an automatic system for translating text from one language to another language without human intervention. They play an important role in the field of entertainment, sports, education, offices, tourism, communication, medical, information technology, research etc. Few real time examples where machine translation plays a very important role are cross-lingual question-answering, multilingual chat sessions, talking translation applications, e-mail and website translations. The above stated are just a few of the modern applications of the commercial world.

There are words that do not need to be translated as they remain the same in all the languages like names of person, place, medicines, terms used in sports etc. These entities are known as “Named Entities” and remain the same whatever be the language and conserve their phonetics.

The process of converting any word from one language to another without changing its pronunciation and phonetics is known as Transliteration. In translation transliteration is used for named entities. It is the process of transcribing one character or letter or alphabet of one language to the other language [P.Antony,2011]. E.g., an English word “School” gets transliterated to Hindi as स्कूल and in Telugu as స్కూల్.

In the proposed research work, a system will be developed for transliteration from English to Hindi and Telugu and also from Hindi to Telugu scripts.

2.0. PROBLEMS IN TRANSLITERATION

Transliteration is a part of Natural Language Processing (NLP) and is useful in Cross language information retrieval, Machine translation, Data mining, etc. While translating a sentence from a script (source script) to other script (target script) the named entities should not get translated but they should be transliterated. For example if “Angel” in a document refers to the name of a person then it should remain Angel in all the languages and it should not get translated for example in Hindi to “परी” or in Telugu to దేవదూత.

Not only for named entities but also for general transliteration from one language to another, it is necessary that pronunciation of the word should remain the same. Thus it makes transliteration a trying task since all the languages have different number of alphabets and each alphabet is associated with different phonetic sounds.

In transliteration, the equivalent phonemes / graphemes of the source script are replaced with those of the target script. There are many problems in transliteration due to the writing style of the script, difference in number of vowels and consonants of the script, difference in phonemes of the characters and missing sounds in some scripts etc.

Basic problems in transliteration:

1. As the number of vowels and consonants is not same in all the scripts and their corresponding phonemes also are different, one cannot use character matching directly for transliteration. The Table 1. gives a comparative position for a few languages / scripts.

LANGUAGE	VOWELS	CONSONANTS
HINDI	13	33+3=36
ENGLISH	5	21
TELUGU	18	38

Table1: Number of Vowels and Consonants in few scripts

2. Not all languages have same sounds / phonemes for their characters. These missing sounds in a language are created by digraph (two characters) or trigraph (three characters) i.e., by combining two or three characters of the script. These missing sounds make the transliteration difficult. For example, in English language, some sounds of Hindi are presented by digraphs “ch”, “sh”, “th” etc. [S.Reddy,2009].

Sounds of Hindi character not present in English characters	Equivalent English character
श	Sh (digraph)
च	Ch (digraph)
क्ष	Ksh (trigraph)

Table2: An example of digraph and tri graph

3. Missing sounds in some languages’ pronunciation also creates difficulties in transliteration, e.g., in pronunciation of a Greek word, “Pneumonia” the letter “P” is silent. English and some other languages use words with origins in Latin / Greek languages. When these languages use words with some silent characters, it becomes difficult to judge which pronunciation technique to use? So origin of the word is an important aspect to be kept in view for transliteration.

4. Sometimes in one language a single character represents a specific sound but the same character transliterated in other language may represent more than one sounds. For example in English letter “T” is equivalent to letter “त” and “ट” letter “D” is equivalent to “द” and “ड” of Hindi.

5. Sometimes the phoneme of a character changes depending upon its surrounding characters. The character or set of characters is pronounced differently depending on the words with which these are used. For example in English “OO” is pronounced differently in “BLOOM”, “BOOK”, “COORDINATOR” etc. “CH” is pronounced differently in “CHARACTER”, “CHEF” and “CHARM”.

Characters	Different pronunciations of same set of characters
OO	Bloom vs. Book vs. Coordinator vs. flood vs. Poor vs. door
Cha	Character vs. Charm vs. Chat Vs. Chalk

Table3: Different pronunciations of same set of characters

6. In some words for example in “scheme” phonemes of “s” and “ch” are used separately while in “schedule” phoneme of “sch” is used.

Phoneme combination	Word
Phoneme of ‘**S’** + phoneme of ‘**ch’**	Scheme
Phoneme of ‘**sch’** together	Schedule

Table 4: Different pronunciation based on character combinations

2.1. Approaches of transliteration

Machine transliteration can be broadly divided into two categories - Rule Based Approach and Statistical Approach.

Rule based approach and Statistical approach: Rule based approach is on the basis of linguistic rules. To formulate these rules one requires a good command over both the languages. V. Goyal et.al. used approximately 50 rules for Hindi to Punjabi machine transliteration [V.Goyal,2009].

Statistical approaches use statistical methods, which include law of probabilities to get the transliterated text. In this method generally the language model is trained with a set of some predefined transliterated text to transliterate between the source and target languages.

Some models of Statistical Approach are as under:

a. <u>Noisy Channel Model:</u>

When a message is created from a source in a human language and it is encoded and transmitted to the receiver through some channel then in that process of transmission some noise gets added to the message. So on the receiver side the encoded message may contain error due to the noise in the transmission channel.

Suppose the original message is “e” and the final / decoded message is “f”. In the given final message we would like to find the original message e by following formula:

$$e' = \arg\max_e P(e|f)$$

If we have error free transmission then by examining a large corpus of message we can construct probability language model P(e), and by examining large corpus of decoded message having noise we can find probability model P(f).

If we know the reason of error in transmission a probability model P(f|e) of the channel can be constructed

By using Baye's law:

$$P(e|f) = \frac{P(f|e)P(e)}{P(f)}$$

so,

$$e' = \arg\max_e P(e|f) = \arg\max_e \frac{P(f|e)P(e)}{P(f)}$$

As we are finding arg max function of e so we can remove P(f) from the denominator ,[Noisy Channel]

$$e' = \arg\max_e P(e|f) = \arg\max_e P(f|e)P(e)$$

In Noisy Channel Model for transliteration, we want to find a transliterated word in target script T' for which probability, P(T|S) is maximum. Where T is the word in target script and S is the word in source script [T.Sherif,2007],

$$T' = \arg\max_T \frac{P(S|T)P(T)}{P(S)}$$

$$T' = \arg\max_T P(S|T)P(T)$$

b. **Hidden Markov Model (HMM)**:

A Hidden Markov Model (HMM) is a sequence of random variables, such that the distribution of these variables depends only on the (hidden) state of an associated Markov chain.

A Hidden Markov Model (HMM) consists of the following:

An alphabet $\Sigma = \{b_1, b_2, \cdots, b_M\}$, a set of states $Q = \{1, 2, \cdots, K\}$.

Transition probabilities between any two states: a_{ij} = the transition probability from state i to j, and for a given state $a_{i1}+a_{i2}+\ldots\ldots\ldots a_{ik}=1$, for all $1 \leq i \leq K$

Start probabilities a_{0i} for all $1 \leq i \leq K$.

Emission probabilities for each state: $e_i(b)$ is the probability of emitting b in state i. We have $e_i(b) = P(x_t = b|\pi_t = i)$

Hidden Markov Model In Tagging: To map a sentence $x_1\ldots\ldots x_n$ to a tag sequence $y_1\ldots\ldots y_n$, is often referred to as a sequence labeling problem, or a tagging problem.

Let $X=x_1,x_2,x_3\ldots\ldots\ldots x_n$ be the input sentence and let $Y=y_1,y_2,y_3\ldots\ldots\ldots y_n$ be the tag sequence.

Joint distribution over word sequence paired with tag sequence $p(x_1\ x_2\ldots\ldots x_n,\ y_1\ y_2\ldots\ldots\ldots y_n)$

$$f(x) = \arg\max_{y_1\ldots\ldots y_n}\ p(x_1x_2.....x_n, y_1y_2....y_n)$$

Thus for any input $x_1 \ldots x_n$, we take the highest probability tag sequence as the output from the model.

Trigram HMMs: A trigram HMM consists of a finite set V of possible words, and a finite set K of possible tags, with the following parameters.

A trigram parameter $q(s \mid u,v)$ for any $s \in K \cup \{STOP\}$, $u, v \in K \cup \{*\}$

A conditional probability or emission parameter $e(x \mid s)$ for any $s \in K$, $x \in V$

Let S be the tag-sequence pairs $< x_1.....x_n, y_1.....y_n >$ such that $n \geq 0$, $x_i \in V$ for $i = 1 \ldots n$, $y_i \in K$ for $i = 1 \ldots n$, and y_{n+1} = STOP.

$$p(x_1.....x_n, y_1.....y_n) = q(stop \mid y_{n-1,}y_n)\prod_{i=1}^{n} q(y_i \mid y_{i-2,}y_{i-1})\prod_{i=1}^{n} e(x_i \mid y_i)$$

y0 = y–1 = *

$$p(x_1.....x_n, y_1.....y_n) = \prod_{i=1}^{n+1} q(y_i \mid y_{i-2,}y_{i-1})\prod_{i=1}^{n} e(x_i \mid y_i)$$

$$p(x_1.....x_n, y_1.....y_n) = q(stop \mid y_{n-1,}y_n)\prod_{i=1}^{n} q(y_i \mid y_{i-2,}y_{i-1})\prod_{i=1}^{n} e(x_i \mid y_i)$$

$$f(x) = \arg\max_{y_1\ldots\ldots y_n}\ p(x_1x_2.....x_n, y_1y_2....y_n)$$

For decoding or finding the highest probability tag sequence dynamic programming algorithm called Viterbi Algorithm is used.[HMM1],[HMM2]

In transliteration when a word sequence S in the source script is to be mapped with transliterated word sequence T in the target script, HMM gives the joint probability P(S,T). [M.collins]

$S=s_1,s_2$………..s_n; $T=t_1,t_2$………..t_n; q is a trigram parameter; and e is conditional probability or emission probability.

$$p(s_1 \ldots\ldots s_n, t_1 \ldots\ldots t_n) = \prod_{i=1}^{n+1} q(t_i | t_{i-2}, t_{i-1}) \prod_{i=1}^{n} e(s_i | t_i)$$

$$T' = \arg\max_T p(s_1 \ldots\ldots s_n, t_1 \ldots\ldots t_n)$$

As the Markov Chain is hidden in the q term it is called a Hidden Markov Model.

c. Maximum Entropy Model

Entropy is a measure of uncertainty of a distribution. MaxEnt model prefers the most uniform models that satisfy any given constraint.

Maximum entropy model is a probabilistic, discriminative classifier which computes the conditional probability of a class y given an observation x i.e. P(y|x).This conditional probability is built using the principle of Maximum entropy.

In the absence of constraints, a uniform probability is assumed for any given class. As we gain constraints (e.g. through training data), the model is modified such that it supports the constraint we have seen but keeps a uniform probability for unseen hypotheses. “Constraint” is given to the MaxEnt model through the use of feature functions. Feature functions provide a numerical value given an observation and weights on these feature functions determine how much a particular feature contributes to a choice of label. In NLP applications, feature functions are often built around words or spelling features in the text.

The MaxEnt model for k competing classes

$$P(y\,|\,x) = \frac{\exp \sum_{i} \lambda_i s_i(x, y)}{\exp \sum_{k} \sum_{i} \lambda_i s_i(x, y_k)}$$

Each feature function s(x,y) is defined in terms of the input observation (x) and the associated label (y) Each feature function has an associated weight (λ), feature functions for a maxEnt model associate a label and an observation. In an NLP application, feature functions might be based on labels (e.g. POS tags) and words in the text.[MaxEnt]

In transliteration if s is a word in source script, t is word in target script, f_i is a feature function and λ_i is a weight associated with the feature function, then according to the MaxEnt model:

$$p(t|s, \lambda) = \frac{\exp \sum_i \lambda_i f_i(t,s)}{Z(t)}$$

Where, Z (t) is the normalization function.

Statistical Tools like Moses and Giza++ are also used for implementing the above four methods. A brief description of these tools is given below:

Moses

Moses is a statistical machine translation system that allows us to automatically train translation models for any language pair. It uses “Phrase based” and “Tree based” translation Models. It also features “Factored translation Models”. [Moses]

Giza++

GIZA++ is an extension of the program GIZA. It is used for word alignments. [Giza]

The rule based approach and statistical approach can be divided further into few more categories based on the method used in transliteration i.e., character matching, phoneme matching, grapheme (letter) matching and hybrid approach. These are represented diagrammatically below:

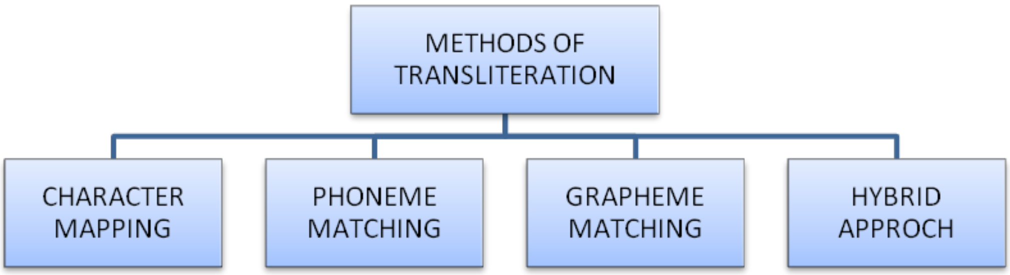


Fig1: Approaches for transliteration

i. Character mapping approach:

Under this approach, the characters of source script are mapped to those of the target script on the basis of pronunciation. Character mapping does not give very good results as the pronunciation of characters and the total number of character varies from script to script. To improve the results other methods have to be used with simple character matching. In a paper, Goyal et. al. used character mapping as the base rule for the Hindi-Punjabi machine transliteration and then added some complex rules for transliteration [V.Goyal,2009].

VOWEL MATCHING	
Hindi	**Telugu**
अ	అ
आ	ఆ

Table5: An Example of Character Matching With Respect To Sound*

ii. Phoneme Based Approach:

This approach defines the relation and correspondence between the phonemes of the source and target script. An alignment of the phoneme for the characters of source script to the phoneme of the target script is done using different methods. I. Kang et.al. used multiple unbounded phoneme chunks for English-Korean transliteration [I.Kang,2000].

English Word	**Equivalent Phoneme Based Segmentation**	**Equivalent Phoneme In Hindi**	**Equivalent Word**
Book	b|ù|k	ब| उ | क	बुक

Table6: An example of phoneme matching for English to Hindi transliteration

iii. Grapheme Based Approach:

This approach defines the relation and correspondence between the graphemes of the source and target scripts. Different methods are used for alignment of the grapheme for the characters of source script with grapheme of the target script. Y. Jia et al. used transliteration as Statistical Machine Translation problem. They used Noisy channel model for grapheme based machine transliteration for English to Chinese machine transliteration [Y.Jia,2009].

English word	Equivalent grapheme based segmentation	Equivalent grapheme in Hindi	Equivalent word
Book	b|oo|k	ब| उ | क	बुक
Put	P|u|t	प|उ|ट or प|उ|त ??	पुट or पुत

Table7: An example of grapheme matching for English to Hindi transliteration

iv. Hybrid Approach

This approach uses the phoneme as well as grapheme of the source and the target scripts to give us a better transliteration model as compared to grapheme or phoneme based approaches.

English word	Equivalent grapheme based segmentation	Equivalent phoneme	Equivalent grapheme in Hindi	Equivalent word
Book	b|oo|k	b|ù|k	ब|उ| क	बुक
Could	c|ou|ld	k|ù|d	क|उ|ड	कुड

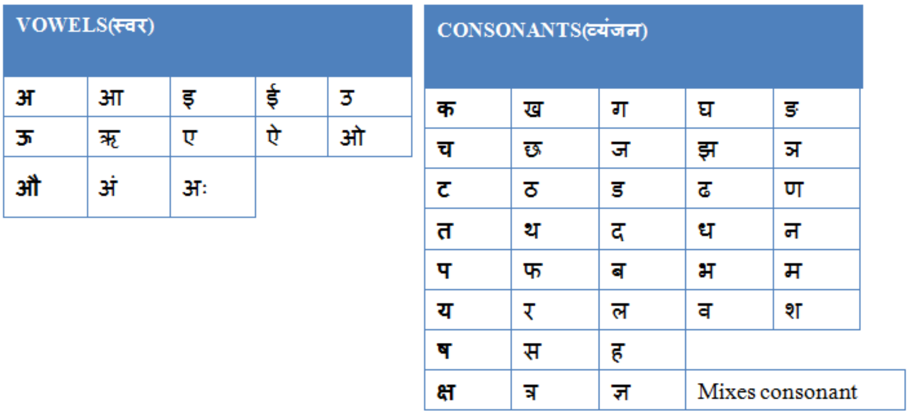
Table8: An example of hybrid approach for English to Hindi transliteration

3.0. IMPORTANT FEATURES OF HINDI, TELUGU & ENGLISH LANGUAGES

3.1. HINDI

In India, Hindi is the national language and is also one of the official languages. Hindi has been considered to have got its name from the Persian word **Hind. Hind** means: 'land of the Indus River'. Turks invaded Punjab and Gangetic plains in the early 11th century gave the name for the language of the region **Hindi** meaning 'language of the land of the Indus River'. Devanagari script is used in writing Modern Hindi. Devanagari is made up of two Sanskrit words: **Deva** ie. ‘God’, & second part **Nagari**, meaning ‘of urban origin’. Devanagari has its origin in Brahmi script.[Hindi]

In Devnagari script, there are 13 vowels and 33 consonants and 3 mixed consonants. Apart from this, each consonant has a half consonant.



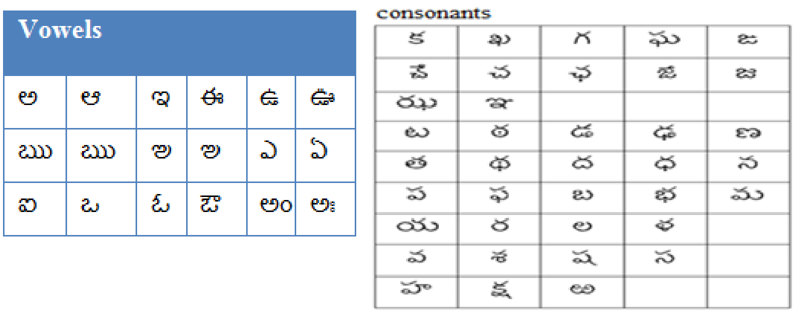
VOWELS(स्वर)				
अ	आ	इ	ई	उ
ऊ	ऋ	ए	ऐ	ओ
औ	अं	अः		

CONSONANTS(व्यंजन)				
क	ख	ग	घ	ङ
च	छ	ज	झ	ञ
ट	ठ	ड	ढ	ण
त	थ	द	ध	न
प	फ	ब	भ	म
य	र	ल	व	श
ष	स	ह		
क्ष	त्र	ज्ञ	Mixes consonant	

Fig.2. Hindi Vowels and consonants

3.2. TELUGU

Telugu is a form of Dravidian language. It is the only language predominantly spoken in more than one Indian state. In Andhra Pradesh and Telangana it is the primary language and in Yanam, it is an official language. Telugu is considered to have been derived from the word: Tenugu (tene = honey, agu = is) meaning sweet as honey. Telugu has 18 vowels and 38 consonants.[Telugu]



Vowels					
అ	ఆ	ఇ	ఈ	ఉ	ఊ
ఋ	ౠ	ఌ	ౡ	ఎ	ఏ
ఐ	ఒ	ఓ	ఔ	అం	అః

consonants

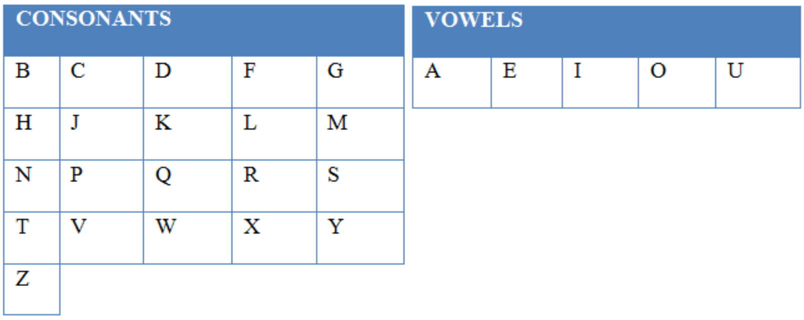
క	ఖ	గ	ఘ	ఙ
చ	ౘ	ఛ	జ	ౙ
ఝ	ఞ			
ట	ఠ	డ	ఢ	ణ
త	థ	ద	ధ	న
ప	ఫ	బ	భ	మ
య	ర	ల	ళ	
వ	శ	ష	స	
హ	క్ష	ఱ		

Fig.3. Telugu Vowels and consonants

3.3. ENGLISH

English is West Germanic language which originated on the lands of England. Now English is a global language and official language for 60 sovereign states.

Modern English is considered to have been derived from Old English, meaning ‘pertaining to the Angles (Engle)’. It was the Germanic tribe in the 5th century. Apart from Angles, Jutes and Saxons were other tribes who lived in Old England, but since the Angles’ language was the first to be written down the word “English” were framed. [English]



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

VOWELS				
A	E	I	O	U

Fig.4. English Vowels and consonants

4.0. LITERATURE SURVEY

[G.S.Josan,2011] - In their paper on Punjabi to Hindi machine transliteration, authors first used a base line method as a character to character matching approach and then compared it with a statistical method for transliteration. They used a Noisy channel model for the purpose. They also concluded that their system can be improved by using some tuning in the language model in terms of alignment heuristics, maximum phrase length etc. and by defining a better syllable similarity score.

[S.Reddy,2009] - In their paper, authors presented a substring based transliteration model and used conditional random fields (CRF) sequential model which use substrings as the basic token unit and pronunciation data as the token level features. They considered source and target language strings as non-overlapping substring sequences. For alignment they have used Giza++ toolkit. They trained the system for English to Hindi, English to Tamil and English to Kannada transliteration and got accuracy of 41.8%, 43.5% and 36.3% respectively.

[T.Rama,2009] - In this paper, authors considered transliteration as a phrase based translation problem for English to Hindi transliteration and used Moses and Giza++. In case of transliteration, phrases are basically the letters of the words. The authors varied the maximum phrase length from 2-7 and changed the order of language model from 2-8 and observed that on training the language model on 7-gram and using alignment heuristic grow-diag-final gives the best results. They got an accuracy of 46.3%.

[V.B.Sowmya,2009] - In this paper, authors described a transliteration based method for typing Telugu using Roman script. They have used Edit-distance based approach using Levenshtein Distance and considered three Levenshtein distances : Levenshtein distance between the two words, between the consonant sets of the two words and between the vowels set of the two words They have concluded that Levenshtein distance gives good results because of the relation between Levenshtein Distance and nature of typing Telugu using English. They used three databases: general database, countries and place names and person names.

[V.Goyal,2009] - In this paper, authors presented a rule based approach for transliteration from Hindi to Punjabi. With the character level mapping of Hindi and Punjabi the authors define approximately 55 rules for transliteration and got an accuracy of 98%.

[A.Finch,2008] - In this paper, authors used phrase based techniques of machine translation for transliteration of English to Japanese words for speech to speech machine translation system. They expressed transliteration as a character level machine translation problem and achieved correct or phonetically equivalent correct words in approximately 80% of cases.

[H.Surana,2008] - In this paper, transliteration from English to Hindi and English to Telugu is done by authors using mapping and fuzzy string matching. Firstly, authors detected the origin of a word in terms of Indian / Foreign word. For foreign words, they mapped English

Phonemes to letters of Indian Language script. For Indian words, they mapped Latin segments of the words to Indian language letters or to a combination of letters and then used fuzzy string matching for final transliteration and got a precision of 80 % for English-Hindi and 71% for English-Telugu.

[T.Sherif,2007] - In this paper, authors have used a substring based transliteration from Arabic to English text. They implemented the method using dynamic programming and finite stat transducers. They evaluated four approaches - a deterministic mapping algorithm (base line method); a letter based transducer; Viterbi substring decoder with obtained optimal substring length as 6; and substring based transducer with obtained best length of substring as 4. The authors then compared results of all these four methods with a fifth approach, viz., manual transliterator. They concluded that substring based transliteration gives better results.

[P.Pingali,2006] - In this paper cross-language retrieval from Hindi and Telugu to English language was done with translations. Authors also used transliteration for proper names and non- dictionary words. They used phoneme mapping, metaphone algorithm and Levenshtein’s approximate string matching for transliteration.

[J.H.Oh,2002] - In this paper on transliteration of English words to Korean words, authors used phonetic information (phoneme and context) and orthographic information for transliteration. They divided English words into two categories - pure English words and those with Greek origin and found that usually pure English words can be transliterated using phoneme and English words with Greek Origin can be transliterated using character matching. After dividing the words in two categories on the basis of origin (E or G) they converted English phonemes to Korean alphabet. They claimed that, their results show an increment of about 31% in word accuracy in comparison to previous works for transliteration.

Summary: In transliteration statistical techniques give good results and these techniques do not require very good linguistic knowledge of the source and the target language. The way vowels are pronounced in a language affects the efficiency of transliterated results. Origin of the words also plays an important role in transliteration. In papers discussed herein above, reasons for error are the origin of words is not taken into account or the way vowels are pronounced and the transliteration system not giving good results for unseen data and abbreviations.

Good results in transliteration can be achieved by using phrase based statistical approach in combination with any of following three methods / approaches individually or also in group: (a) Substring based approach; (b) Pronunciation scheme of a language; and (c) origin of words.

5.0. PROPOSED WORK

The present research work will be on transliteration from English to Hindi and Telugu and from Hindi to Telugu. A transliteration system from languages like English and Hindi to Telugu will be very useful for Cross-language Information Retrieval, translation, in studying the pronunciation of English and Hindi words for those who can understand English, Hindi and Telugu but can't read English and Hindi and similarly transliteration from English to Hindi will be useful for those who can understand English, and Hindi but can't read English.

In the present Research work we will use Basic Statistical Methods for transliteration from English to Hindi and Telugu and Hindi to Telugu using tools like Moses and Giza++.

As given in literature for other languages substring based statistical methods give better results for transliteration in comparison to base line methods or rule based method which requires good linguistic knowledge of the source language as well as target language. We will consider Transliteration from English to Hindi and Telugu and Hindi to Telugu as a substring based transliteration problem.

We will also consider transliteration as phrase based statistical machine translation problem. Phrase based methods for transliteration is similar to SMT (Statistical Machine Translation) techniques. SMT is smart translation which considers a group of words and their interdependency rather than individual word translation. In SMT method, the model considers group of words as a phrase and then translates from source language to target language and similarly in transliteration if SMT method is applied, the model considers one individual word as a phrase and individual characters as words for proper conversion.

6.0. REFERENCES

[A.Finch,2008] Finch, Andrew, and Eiichiro Sumita, "Phrase-based machine transliteration" in Proceedings of the Workshop on Technologies and Corpora for Asia-Pacific Speech Translation (TCAST), pp. 13-18. 2008.

[G.S.Josan,2011] Josan, Gurpreet Singh, and Jagroop Kaur, "Punjabi to Hindi statistical machine transliteration." International Journal of Information Technology and Knowledge Management 4, no. 2 ,pp. 459-463. 2011

[H.Surana,2008] Surana, Harshit, and Anil Kumar Singh, "A More Discerning and Adaptable Multilingual Transliteration Mechanism for Indian Languages" in IJCNLP, pp. 64-71. 2008.

[I.Kang,2000] Kang, In-Ho, and GilChang Kim, "English-to-Korean transliteration using multiple unbounded overlapping phoneme chunks" in Proceedings of the 18th conference on Computational linguistics-Vol. 1, Assoc. for Computational Linguistics pp. 418-424., 2000.

[J.H.Oh,2002] Oh, Jong-Hoon, and Key-Sun Choi, "An English-Korean transliteration model using pronunciation and contextual rules" in Proceedings of the 19th international conference on Computational linguistics-Vol. 1, Association for Computational Linguistics, pp. 1-7. 2002.

[P.Antony,2011] Antony, P. J and K. P. Soman, "Machine transliteration for Indian languages: A literature survey." International Journal of Scientific & Engineering Research, IJSER 2, pp. 1-8. 2011

[P.Pingali,2006] Pingali, Prasad, and Vasudeva Varma, "Hindi and Telugu to English Cross Language Information Retrieval at CLEF 2006" in Working Notes of Cross Language Evaluation Forum, 2006.

[S.Reddy,2009] Reddy, Sravana, and Sonjia Waxmonsky, "Substring-based transliteration with conditional random fields" in Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration, Association for Computational Linguistics, pp. 92-95. 2009.

[T.Rama,2009] Rama, Taraka, and Karthik Gali, "Modeling machine transliteration as a phrase based statistical machine translation problem" in Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration, Association for Computational Linguistics, pp. 124-127. 2009.

[T.Sherif,2007] Sherif, Tarek, and Grzegorz Kondrak, "Substring-based transliteration" in Annual Meeting of Association for Computational Linguistics, vol. 45, no. 1, pp. 944-951. 2007.

[V.B.Sowmya,2009] Sowmya, V. B., and Vasudeva Varma, "Transliteration based text input methods for telugu" in Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy, Springer Berlin Heidelberg, pp. 122-132., 2009.

[V.Goyal,2009] Goyal, Vishal, and Gurpreet Singh Lehal, "Hindi-Punjabi Machine Transliteration System (For Machine Translation System)." George Ronchi Foundation Journal, Italy 64, no. 1. 2009.

[Y.Jia,2009] Jia, Yuxiang, Danqing Zhu, and Shiwen Yu, "A noisy channel model for grapheme-based machine transliteration" in Proceedings of the 2009 Named Entities

Workshop: Shared Task on Transliteration, Association for Computational Linguistics, pp. 88-91. 2009.

[English] https://en.wikipedia.org/wiki/English_language

[Giza] http://www.statmt.org/moses/?n=Moses.Overview

[Hindi] http://www.bbc.co.uk/languages/other/hindi/guide/alphabet.shtml

[HMM1]http://www.cs.columbia.edu/~mcollins/courses/nlp2011/notes/hmms.pdf

[HMM2] http://robotics.stanford.edu/~serafim/CS262_2008/notes/lecture6.pdf

[M.Collins] http://www.cs.columbia.edu/~mcollins/hmms-spring2013.pdf

[MaxEnt] web.cse.ohio-state.edu/~morrijer/Presentations/cse7881008_jjm.ppt

[moses] http://www.statmt.org/moses/

[Noisy Channel] https://www.youtube.com/watch?v=zjWXLD_ihOc

[Telugu]https://en.wikipedia.org/wiki/Telugu_language & https://telugubasha.net/en/history