

# English to Gujarati Transliteration using Machine Learning

Payal Joshi

Department of Information and Communication Technology,  
Veer Narmad South Gujarat University, Surat, India

**Abstract-** Transliterated search has become a crucial requirement for current search engines and digital libraries. This is because of increase in digitization of native language contents. Substantial amount of work is done on this problem for various Indian languages. No substantial work is done for Gujarati language for transliterated search. Most of the work is done using rule based approach. Machine learning based approach is required so that we can achieve more accuracy and flexibility of adding new combinations for transliterations for Gujarati language. In this paper, Conditional Random Field (CRF) based machine learning approach is used for transliteration of Roman script word to Gujarati language. These transliterations can be expanded with query in order to match and retrieve documents written in Gujarati script.

**CCS Concepts-** Natural Language Processing, Information Retrieval

**Keywords-** Gujarati language transliteration; Information Retrieval

## I. INTRODUCTION

Because of digitization, enormous contents of various Indian language is published online. Searching such contents is difficult for end users because many internet users do not know how to type a query in native script. This is also because of unavailability of language input tools. To search Gujarati language contents, user has to type query in Gujarati language. In other case, user may want to search using Roman script. E.g. User should be able to type “Mahatma Gandhi” in order to get documents related to Mahatma Gandhi written in native language. So queries searched on the web are written in native script or Roman script to find mono lingual contents written in native language.

In such cases, we can transliterate query words and match it with documents. It is a major challenge to match Roman script query words with native language documents. In information retrieval, this makes a wide scope of research for various languages.

Transliterated search is a crucial requirement for current search engines and digital libraries. Good amount of work is done on this problem for various Indian languages. In Gujarati language also work has been done but most of the work is done using rule based approach. Machine learning based approach is required so that we can achieve more accuracy and flexibility of adding new combinations for transliterations in order to cope up with wide variety of vocabulary and spelling variations.

In this paper, machine learning approach is used for transliteration of Roman script word to Gujarati language with the objective to broaden the scope of search for Gujarati language documents by giving user flexibility to search Gujarati documents using Roman script and/or Gujarati script to match mono-lingual Gujarati documents.

These transliterations are then expanded with query in order to match and retrieve documents written in both Gujarati and Roman script. In this paper major focus is given to transliteration of Roman script words to Gujarati. E. g. Query “Mahatma Gandhi” should be expanded as “મહાત્મા ગાંધી” and it retrieves documents containing words Mahatma and/or Gandhi in Gujarati language. Rest of the paper is organized as follows: Section II describes related work. Section III describes our approach of transliteration. Section IV shows results and analysis and section V concludes the paper.

## II. RELATED WORK

Shraddha Patel and Vaibhavi Desai in [1], in FIRE 2014 in Mixed Script IR task have employed combination of bi-gram and tri-gram with rule based approach for transliteration. They have used Hindi as base language for transliteration to Gujarati language.

Royal Denzil Sequiera, Shashank S Rao, and Shambavi B R in [2] have also used rule based tri-gram approach to identify language and dictionary based approach to back transliterate a word to its native script.

Irshad Ahmad Bhat, Vandan Mujadia, Aniruddha Tammewar, Riyaz Ahmad Bhat, and Manish Shrivastava in [3] have used ID3 classifier and Indic-converter for transliteration of Gujarati language.

Substantial work is done in Indian languages for transliteration, especially in Hindi language. H Joshin et. al. in [5] used Viterbi algorithm for transliteration of English to Hindi words. S Gella et. al in [6] have used Indic character mapping for English to Hindi transliteration. P Gupta et. al. in [7] have proposed method to handle Hindi transliteration variations through non-linear dimensionality reduction techniques.

V Singhal et. al. in [8] developed rule based syllabification and rule based approach for English to Hindi transliteration. P Velunkar et. al. in [9] have also adapted rule based approach for language identification as well as transliteration. Sharma et. al. [10] trained a statistical machine translation system for successfully translating English-Hindi named entities using CRF-based approach. They showed 85.79% accuracy and

showed that CRF is best suited for processing Indian languages. [9][10].

For Gujarati language machine learning based approach is required so that we can achieve more accuracy and flexibility of adding new combinations for transliterations.

III. METHODOLOGY

In this section, we describe resources and implementation for transliteration of Roman script Gujarati words to Gujarati script.

a. Resources

For implementation of this methodology following data resources are used:

1. English-Gujarati Transliterated pairs for Gujarati language words.
2. English-Gujarati Transliterated pairs of manually syllabified words for training. Sample data is shown in Figure [1].

a-ka-smaa-t	અ-ક-સ્મા-ત
A-ksha-r-dha-m	અ-ક્ષ-ર-ધા-મ
a-tyaa-re	અ-ત્યા-રે
a-n-ra-dha-r	અ-ન-રા-ધા-ર
a-ne	અ-ને
a-ne-k	અ-ને-ક
a-pe-ksha	અ-પે-ક્ષા
a-bhi-ne-tri	અ-ભિ-ને-ત્રી
A-m-da-va-d	અ-મ-દા-વા-દ
a-m-da-vaa-d-na	અ-મ-દા-વા-દ-ના
a-ma-sta	અ-મા-સ્તા
a-yaa-i	અ-યા-ઈ
an-ti-m	અં-તિ-મ
an-da-je	અં-દા-જે
a-ha-ma-ne	અ-હ-મ-ને

Fig.1: English-Gujarati Transliterated pairs of manually syllabified words used for CRF training.

b. Implementation

Conditional Random Field (CRF) machine learning approach is applied to automatically transliterate Gujarati and English words for which transliterated pair is not found. In this 1000 word pairs of manually syllabified English-Gujarati is used for training the model and 100 words are used for testing. (Words list is downloaded from [4] and more words are also added from different sources, training words are manually syllabified to achieve higher efficiency) We have developed a model for Roman script to Gujarati transliteration.

CRF defines a conditional probability over label sequence Y given a particular observation sequence X. To apply CRFs to transliteration problem, an observation sequence X is a string of source language transliteration units {x1,x2,...,xn} and state sequence Y is the string of target language transliteration units {y1,y2,...,yn}. CRF model is trained using manually syllabified Romanized Gujarati (X) and Gujarati (Y) word fragments as shown in Figure 1. In machine transliteration, conditional random field model can be used to generate the target

language word from a source language word. E.g. Word manushya is modeled in CRF as shown in figure 1. For transliteration, word ‘manushya’ is first syllabified as ma-nu-shya and then it is transliterated as મ-નુ-શ્ય using CRF and back-transliterated syllabified fragments are combined back as મનુશ્ય. Word “manushya” can be modeled in CRF as shown in Figure 2 below.

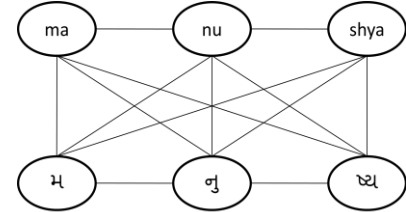


Fig.2: CRF modeling for transliteration

For transliteration of a word from Roman script to Gujarati script, first a query word is syllabified using rule based approach. For this word is broken down considering vowels followed by a consonant as a delimiter as shown in figure- Then it is transliterated using CRF based machine learning, to Gujarati word using model trained before.

These transliterations may then be expanded with query in order to match retrieve documents written in Gujarati script. Figure-3 below shows flowchart of the CRF based machine transliteration for English to Gujarati script.

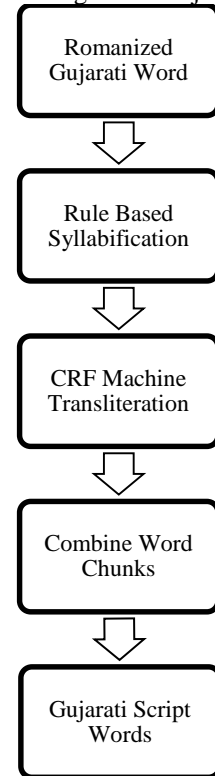
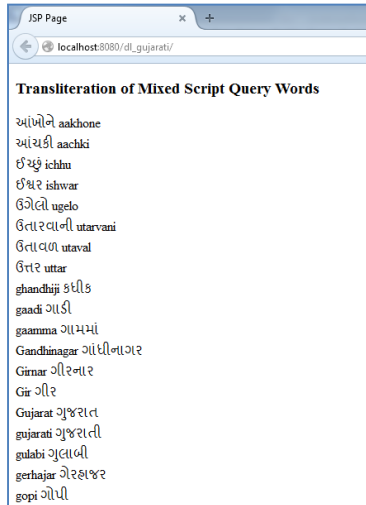


Fig.3: Flowchart of CRF based machine transliteration for English to Gujarati.

#### IV. RESULTS AND ANALYSIS

Sample of output in JSP page is as shown in Figure-4. It consists of query word and its transliterated word. First column of words shows input word and second column shows its transliterated form.



Input Word (Gujarati)	Transliterated Word (Roman)
આખોને	aaakhone
આંચકી	aachki
ઈચ્છું	ichhu
ઈશ્વર	ishwar
ઉગેલો	ugelo
ઉતારવાની	utarvani
ઉતાવળ	utaval
ઉત્તર	uttar
ghandhiji	કંઈક
gaadi	ગાડી
gaamma	ગામમાં
Gandhinagar	ગાંધીનગર
Ginar	ગીરનાર
Gi	ગીર
Gujarat	ગુજરાત
gujarati	ગુજરાતી
gulabi	ગુલાબી
gerhajar	ગેરહજર
gopi	ગોપી

Fig.4: Sample of transliteration result.

Performance of transliteration is as shown in Table 1.

	Number of words
Total Roman Words Tested	100
Correct Transliterations	84
Incorrect Transliterations	16

**Table 1 Transliteration result (Roman script to Gujarati)**

These transliterations may be expanded with query in order to retrieve documents written in both Gujarati and Roman script. E. g. Query “mahatma ગાંધી” is expanded as “મહાત્મા + ગાંધી” and it retrieves documents containing words Mahatma and/or Gandhi in Gujarati language.

For CRF transliteration, first best match is used. But best 5 match are also considered and it is planned to make various transliteration pairs out of it and finding best out of them using edit distance matching. It is required for transliteration to match i, ee, s, sh, shh, a, aa. ળ and ળ with N, ળ and ળ with L.

#### V. CONCLUSION

Transliterated search is a crucial requirement for current search engines and digital libraries. In this paper CRF based machine learning approach is used for transliteration of Roman script word to Gujarati language. These transliterations are then expanded in query in order to retrieve documents written in both Gujarati and Roman script.

This will make the scope of search broad for Gujarati language documents by giving user flexibility to search Gujarati documents using roman script to match mono-lingual Gujarati documents. In future we are going to enhance this work by adding more training data in order to cope up with wide vocabulary of both languages and also going to extend our work by adding more features.

#### VI. REFERENCES

- [1]. Patel, S., and Desai, V. 2014. LIGA and Syllabification Approach for Language Identification and Back Transliteration. *Shared Task Reported by DAICT in FIRE-2014.*
- [2]. Royal, D. S., Rao, S. S., and Shambavi, B. R. 2014. Word-Level Language Identification and Back Transliteration of Romanized Text. *A Shared Task Report by BMSCE in FIRE-2014.*
- [3]. Bhat, I. A., Mujadia, V., Tammewar, A., Bhat, R. A., and Shrivastava, M. 2014. IIIT-H System Submission for FIRE2014 Shared Task on Transliterated Search. *A Shared Task Report in FIRE-2014.*
- [4]. <http://cse.iitkgp.ac.in/resgrp/cnerg/qa/fire13translit/index.html>. As accessed in August 2015.
- [5]. Agarwal, A. 2010. Transliteration involving English and Hindi languages using Syllabification Approach. *M.Tech thesis, Indian Institute of Technology, Bombay.* (Jan. 2010).
- [6]. Gella, S., Sharma, J., and Bali, K. 2013. Query word labeling and Back Transliteration for Indian Languages. *A Shared task in FIRE 2013.*
- [7]. Gupta, P., Rosso, P., and Banchs, R. E. 2013. Encoding transliteration variation through dimensionality reduction. *A Shared Task on Transliterated Search in FIRE-2013.*
- [8]. Singhal, V., and Tyagi, N. 2015. A Hybrid Approach of English – Hindi Named – Entity Transliteration. *International Journal of Advanced Technology in Engineering and Science.* 3, 2 (Feb. 2015), 580-587, ISSN: 2348 – 7550.
- [9]. Verulkar, P., Balabantray, R. C., and Chakrapani, R. A. 2015. Transliterated Search on Hindi Lyrics. *International Journal of Computer Application.* 121, 1 (Jul. 2015), 32-37, ISSN: 0975 – 8887.
- [10]. Sharma, S., Bora, N., and Halder, M. 2012. English-Hindi Transliteration Using Statistical Machine Translation in Different Notation. *International Conference on Computing and Control Engineering.* (Apr. 2012), ISBN: 978-1-4675-2248-9.