Hindi Urdu Machine Transliteration using Finite-state Transducers

Muhammad Ghulam Abbas Malik, Christian Boitet, Pushpak Bhattacharyya

Hindi Urdu Machine Transliteration using Finite-state Transducers

M G Abbas Malik **Christian Boitet**
GTALP, Laboratoire d'Informatique Grenoble
Université Joseph Fourier, France
abbas.malik@imag.fr,
Christian.Boitet@imag.fr

Pushpak Bhattacharyya
Dept. of Computer Science and Engineering,
IIT Bombay, India
pb@cse.iitb.ac.in

Abstract

Finite-state Transducers (FST) can be very efficient to implement inter-dialectal transliteration. We illustrate this on the Hindi and Urdu language pair. FSTs can also be used for translation between surface-close languages. We introduce UIT (universal intermediate transcription) for the same pair on the basis of their common phonetic repository in such a way that it can be extended to other languages like Arabic, Chinese, English, French, *etc.* We describe a transliteration model based on FST and UIT, and evaluate it on Hindi and Urdu corpora.

1 Introduction

Transliteration is mainly used to transcribe a word written in one language in the writing system of the other language, thereby keeping an approximate phonetic equivalence. It is useful for MT (to create possible equivalents of unknown words) (Knight and Stall, 1998; Paola and Sanjeev, 2003), cross-lingual information retrieval (Pirkola et al, 2003), the development of multilingual resources (Yan et al, 2003) and multilingual text and speech processing. Inter-dialectal translation without lexical changes is quite useful and sometimes even necessary when the dialects in question use different scripts; it can be achieved by transliteration alone. That is the case of HUMT (Hindi-Urdu Machine Transliteration) where each word has to be transliterated from Hindi to Urdu and *vice versa*, irrespective of its type (noun, verb, *etc.* and not only proper noun or unknown word).

"One man's Hindi is another man's Urdu" (Rai, 2000). The major difference between Hindi and Urdu is that the former is written in Devanagari script with a more Sanskritized vocabulary and the latter is written in Urdu script (derivation of Persio-Arabic script) with more vocabulary borrowed from Persian and Arabic. In contrast to the transcriptional difference, Hindi and Urdu share grammar, morphology, a huge vocabulary, history, classical literature, cultural heritage, *etc.* Hindi is the National language of India with 366 million native speakers. Urdu is the National and one of the state languages of Pakistan and India respectively with 60 million native speakers (Rahman, 2004). Table 1 gives an idea about the size of Hindi and Urdu.

	Native Speakers	2nd Language Speakers	Total
Hindi	366,000,000	487,000,000	853,000,000
Urdu	60,290,000	104,000,000	164,290,000
Total	426,290,000	591,000,000	**1,017,000,000**

Table 1: Hindi and Urdu speakers

Hindi and Urdu, being varieties of the same language, cover a huge proportion of world's population. People from Hindi and Urdu communities can understand the verbal expressions of each other but not the written expressions. HUMT is an effort to bridge this scriptural divide between India and Pakistan.

Hindi and Urdu scripts are briefly introduced in section 2. Universal Intermediate Transcription (UIT) is described in section 3, and UIT mappings for Hindi and Urdu are given in section 4. Contextual HUMT rules are presented and discussed in section 5. An HUMT system implementation and its evaluation are provided in section 6 and 7. Section 8 is on future work and conclusion.

Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 537–544
Manchester, August 2008

2 HUMT

There exist three languages at the border between India and Pakistan: Kashmiri, Punjabi and Sindhi. All of them are mainly written in two scripts, one being a derivation of the Persio-Arabic script and the other being Devanagari script. A person using the Persio-Arabic script cannot understand the Devanagari script and *vice versa*. The same is true for Hindi and Urdu which are varieties or dialects of the same language, called *Hindustani* by Platts (1909).

PMT (Punjabi Machine Transliteration) (Malik, 2006) was a first effort to bridge this scriptural divide between the two scripts of Punjabi namely Shahmukhi (a derivation of Perio-Arabic script) and Gurmukhi (a derivation of Landa, Shardha and Takri, old Indian scripts). HUMT is a logical extension of PMT. Our HUMT system is generic and flexible such that it will be extendable to handle similar cases like Kashmiri, Punjabi, Sindhi, *etc*. HUMT is also a special type of machine transliteration like PMT.

A brief account of Hindi and Urdu is first given for unacquainted readers.

2.1 Hindi

The Devanagari (literally "godly urban") script, a simplified version of the alphabet used for Sanskrit, is a left-to-right script. Each consonant symbol inherits by default the vowel sound [ə]. Two or more consonants may be combined together to form a cluster called Conjunct that marks the absence of the inherited vowel [ə] between two consonants (Kellogg, 1872; Montaut, 2004). A sentence illustrating Devanagari is given below:

हिन्दी हिन्दुस्तान की क़ौमी ज़ुबान है.

[hɪn̪di hɪn̪d̪ustɑn ki qɔmi zubɑn hæ]
(Hindi is the national language of India)

2.2 Urdu

Urdu is written in an alphabet derived from the Persio-Arabic alphabet. It is a right-to-left script and the shape assumed by a character in a word is context-sensitive, *i.e.* the shape of a character is different depending on whether its position is at the beginning, in the middle or at the end of a word (Zia, 1999). A sentence illustrating Urdu is given below:

اردو پاکستان کی قومی زبان ہے۔

[ʊrd̪u pɑkɪstɑn ki qɔmi zubɑn hæ]
(Urdu is the National Language of Pakistan.)

3 Universal Intermediate Transcription

UIT (Universal Intermediate Transcription) is a scheme to transcribe texts in Hindi, Urdu, Punjabi, *etc.* in an unambiguous way encoded in ASCII range 32 – 126, since a text in this range is portable across computers and operating systems (James 1993; Wells, 1995). SAMPA (Speech Assessment Methods Phonetic Alphabet) is a widely accepted scheme for encoding the IPA (International Phonetic Alphabet) into ASCII. It was first developed for Danish, Dutch, French, German and Italian, and since then it has been extended to many languages like Arabic, Czech, English, Greek, Hebrew, Portuguese, Russian, Spanish, Swedish, Thai, Turkish, *etc.*

We define UIT as a logical extension of SAMPA. The UIT encoding for Hindi and Urdu is developed on the basis of rules and principles of SAMPA and X-SAMPA (Wells, 1995), that cover all symbols on the IPA chart. Phonemes are the most appropriate invariants to mediate between the scripts of Hindi, Punjabi, Urdu, *etc.*, so that the encoding choice is logical and suitable.

4 Analysis of Scripts and UIT Mappings

For the analysis and comparison, scripts of Hindi and Urdu are divided into different groups on the basis of character types.

4.1 Consonants

These are grouped into two categories:

Aspirated Consonants: Hindi and Urdu both have 15 aspirated consonants. In Hindi, 11 aspirated consonants are represented by separate characters *e.g.* ख [k^h], भ [b^h], *etc.* The remaining 4 consonants are represented by combining a simple consonant to be aspirated and the conjunct form of HA ह[h], *e.g.* ल [l] + ् + ह [h] = ल्ह [l^h].

In Urdu, all aspirated consonants are represented by a combination of a simple consonant to be aspirated and Heh Doachashmee (ھ) [h], *e.g.* ک [k] + ھ [h] = کھ [k^h], ب [b] + ھ [h] = بھ [b^h], ل [l] + ھ [h] = لھ [l^h], *etc.*

The UIT mapping for aspirated consonants is given in Table 2.

Hindi	Urdu	UIT	Hindi	Urdu	UIT
भ	بھ [b^h]	b_h	ह्र	رھ [r^h]	r_h
फ	پھ [p^h]	p_h	ढ़	ڑھ [$ɽ^h$]	r`_h
थ	تھ [$t̪^h$]	t_d_h	ख	کھ [k^h]	k_h
ठ	ٹھ [$ʈ^h$]	t`_h	घ	گھ [g^h]	g_h
झ	جھ [$ʤ^h$]	d_Z_h	ल्ह	لھ [l^h]	l_h

छ	چھ [ʧʰ]	t_S_h	म्ह	مھ [mʰ]	m_h
ध	دھ [ɖʰ]	d_d_h	न्ह	نھ [nʰ]	n_h
ढ	ڈھ [ɖʰ]	d`_h			

Table 2: Hindi Urdu aspirated consonants

Non-aspirated Consonants: Hindi has 29 non-aspirated consonant symbols representing 28 consonant sounds as both SHA (श) and SSA (ष) represent the same sound [ʃ]. Similarly Urdu has 35 consonant symbols representing 27 sounds as multiple characters are used to represent the same sound *e.g.* Heh (ح) and Heh-Goal (ہ) represent the sound [h] and Theh (ث), Seen (س) and Sad (ص) represent the sound [s], *etc.*

UIT mapping for non-aspirated consonants is given in Table 3.

Hindi	Urdu	UIT	Hindi	Urdu	UIT
ब	ب [b]	b	स	ص [s]	s2
प	پ [p]	p	ज़	ض [z]	z2
त	ت [ʈ]	t_d	त	ط [ʈ]	t_d1
ट	ٹ [ʈ]	t`	ज़	ظ [z]	z3
स	ٹ [s]	s1	-	ع [ʔ]	?
ज	ج [ʤ]	d_Z	ग़	غ [ɣ]	X
च	چ [ʧ]	t_S	फ़	ف [f]	f
ह	ح [h]	h1	क़	ق [q]	q
ख़	خ [x]	x	क	ک [k]	k
द	د [ɖ]	d_d	ग	گ [g]	g
ड	ڈ [ɖ]	d`	ल	ل [l]	l
ज़	ذ [z]	z1	म	م [m]	m
र	ر [r]	r	न	ن [n]	n
ड़	ڑ [ɽ]	r`	व	و [v]	v
ज़	ز [z]	z	ह	ہ [h]	h
ज़	ژ [ʒ]	Z	य	ی [j]	j
स	س [s]	s	त	ۃ [ʈ]	t_d2
श	ش [ʃ]	S	ण	- [ɳ]	n`
ष	ش [ʃ]	S1	ं	ں [ŋ]	~

Table 3: Hindi Urdu non-aspirated consonants

4.2 Vowels

Hindi has 11 vowels and 10 of them have nasalized forms. They are represented by 11 independent vowel symbols *e.g.* आ [ɑ], ऊ [u], औ [ɔ], *etc.* and 10 dependent vowel symbols *e.g.* ा [ɑ], ू [u], ौ [ɔ], *etc.* called *maatraas*. When a vowel comes at the start of a word or a syllable, the independent form is used; otherwise the dependent form is used (Kellogg, 1872; Montaut, 2004).

Urdu contains 10 vowels and 7 of them have nasalized forms (Hussain, 2004; Khan, 1997). Urdu vowels are represented using four long vowels (Alef Madda (آ), Alef (ا), Vav (و) and Choti Yeh (ی)) and three short vowels (Arabic Fatha – Zabar ـَ, Arabic Damma – Pesh ـُ and Arabic Kasra – Zer ـِ). Vowel representation is context-sensitive in Urdu. Vav (و) and Choti Yeh (ی) are also used as consonants.

Hamza (ء) is a place holder between two successive vowel sounds, *e.g.* in کمائی [kəmɑi] (earning), Hamza (ء) separates the two vowel sounds Alef (ا) [ɑ] and Choti Yeh (ی) [i]. Noon-ghunna (ں) is used as nasalization marker. Analysis and mapping of Hindi Urdu vowels is given in Table 5.

4.3 Diacritical Marks

Urdu contains 15 diacritical marks. They represent vowel sounds, except Hamza-e-Izafat ـٔ and Kasr-e-Izafat ـِ that are used to build compound words, *e.g.* اِدارۂ سائنس [ɪɖərəhɪsaɪns] (Institute of Science), تاریخ پیدائش [tarixɪpedaɪʃ] (date of birth), *etc.* Shadda ـّ is used to geminate a consonant *e.g.* ربّ [rəbb] (God), اچّھا [əʧʧʰɑ] (good), *etc.* Jazm ـْ is used to mark the absence of a vowel after the base consonant (Platts, 1909). In Hindi, the conjunct form is used to geminate a consonant. Urdu diacritical marks mapping is given in Table 4.

Hindi	Urdu	UIT	Hindi	Urdu	UIT
-	◌ [ə]	@	◌ा	◌ [ɑ]	A
◌ि	◌ [ɪ]	I	◌न	◌ [ən]	@n
◌ु	◌ [ʊ]	U	◌ुन	◌ [ʊn]	Un
◌ृ	◌ [u]	u	◌िन	◌ [ɪn]	In
◌ी	◌ [i]	i			

Table 4: Diacritical Marks of Urdu

Diacritical marks are present in Urdu but sparingly used by people. They are very important for the correct pronunciation and understanding the meanings of a word. For example,

یہ سڑک بہت چوڑی ہے۔

[je səɽək bʊhət ʧɔɽi hæ] (This is a wide road.)

میری چوڑی سرخ ہے۔

[meri ʧʊɽi sʊrəx hæ] (My bangle is red.)

In the first sentence, the word چوڑی is pronounced as [ʧɔɽi] (wide) and in the second, it is

pronounced as [ʧuɽi] (bangle). There should be Zabar (ـَ) and Pesh (ـُ) after Cheh (چ) in above words and correct transcriptions are چَوڑی (wide) and چُوڑی (bangle). Thus diacritical marks are essential for removing ambiguities, natural language processing and speech synthesis.

Vowel	Urdu	Hindi (UIT)
ə	It is represented by Alef (ا) + Zabar ـَ at the start of a word *e.g.* اَب [əb] (now) and by Zabar ـَ in the middle of a word respectively *e.g.* رَبّ [rəbb] (God). It never comes at the end of a word.	अ (@)
ɑ	It is represented by Alef Madda (آ) at the start of a word *e.g.* آدمی [ɑd̪mi] (man) and by Alef (ا) or Alef Madda (آ) in the middle of a word *e.g.* جانا [ʤɑnɑ] (go), بلآخر [bɪlɑxər] (at last). At the end of a word, it is represented by Alef (ا). In some Arabic loan words, it is represented by Choti Yeh (ی) + Khari Zabar ٰ at the end of a word *e.g.* اعلیٰ [ə?lɑ] (Superior) and by Khari Zabar ٰ in the middle of a word *e.g.* الٰہی [ɪlɑhi] (God).	आ or ा (A)
e	It is represented by Alef (ا) + Choti Yeh (ی) at the start of a word *e.g.* ایثار [esɑr] (sacrifice), ایک [ek] (one), *etc.* and by Choti Yeh (ی) or Baree Yeh (ے) in the middle of a word *e.g.* میرا [merɑ] (mine), اندھیرا [ənd̪^{h}erɑ] (darkness), بےگھر [beg^{h}ər] (homeless) *etc.* At the end of a word, It is represented by Baree Yeh (ے) *e.g.* سارے [sɑre] (all).	ए or े (e)
æ	It is represented by Alef (ا) + Zabar ـَ + Choti Yeh (ی) at the start of a word *e.g.* ایسہ [æh] (this) and by Zabar ـَ + Choti Yeh (ی) in the middle of a word *e.g.* مَیل [mæl] (dirt). At the end of a word, it is represented by Zabar ـَ + Baree Yeh (ے) *e.g.* ہَے [hæ] (is).	ऐ or ै (ˌ)
ɪ	It is represented by Alef (ا) + Zer ـِ at the start of a word *e.g.* اِس [ɪs] (this) and by Zer ـِ in the middle of a word *e.g.* بارش [bɑrɪʃ] (rain). It never comes at the end of a word. At the end of a word, it is used as Kasr-e-Izafat to build compound words.	इ or ि (I)
i	It is represented by Alef (ا) + Zer ـِ + Choti Yeh (ی) at the start of a word *e.g.* ایمان [imɑn] (belief) and by Zer ـِ + Choti Yeh (ی) in the middle or at the end of a word *e.g.* امیری [ɑmiri] (richness), قریب [qərib] (near), *etc.*	ई or ी (i)
ʊ	It is represented by Alef (ا) + Pesh ـُ at the start of a word *e.g.* اُدّھر [ʊd̪d̪^{h}ər] (there) and by Pesh ـُ in the middle of a word *e.g.* مُلّ [mʊll] (price). It never comes at the end of a word.	उ or ु (U)
u	It is represented by Alef (ا) + Pesh ـُ + Vav (و) at the start of a word *e.g.* اُونگھتا [ũg^{h}ə̤tɑ] (dozzing) and by Pesh ـُ + Vav (و) in the middle or at the end of a word *e.g.* صُورت [surə̤t] (face), ترازُو [ʈ̤ərɑzu] (physical balance), *etc.*	ऊ or ू (u)
o	It is represented by Alef (ا) + Vav (و) at the start of a word *e.g.* اوچھا [oʧ^{h}ɑ] (nasty) and by Vav (و) in the middle or at the end of a word *e.g.* ہولی [holi] (slowly), کہو [kəho] (say), *etc.*	ओ or ो (o)
ɔ	It is represented by Alef (ا) + Zabar ـَ + Vav (و) at the start of a word *e.g.* اَوٹ [ɔʈ̤] (hindrance) and by Zabar ـَ + Vav (و) in the middle or at the end of a word *e.g.* مَوت [mɔʈ̤] (death).	औ or ौ (O)
r̥	It is represented by a consonant symbol Reh (ر) [r] as this vowel is only present in Sanskrit loan words. It is almost not used in modern standard Hindi. It is not present in Urdu as it is used only in Sanskrit loan words.	ऋ or ृ (r1)
Note: In Hindi, Nasalization of a vowel is done by adding Anunasik (ँ) or Anusavar (ं) after the vowel. Anusavar (ं) is used when the vowel graph goes over the upper line; otherwise Anunasik (ँ) is used (Kellogg, 1872; Montaut, 2004). In UIT, ~ is added at end of UIT encoding for nasalization of all above vowels except the last one that do not have a nasalized form.		

Table 5: Analysis and Mapping of Hindi Urdu Vowels

5 HUMT Rules

In this section, UIT mappings of Hindi Urdu alphabets and contextual rules that are necessary for Hindi-Urdu transliteration are discussed.

5.1 UIT Mappings

UIT mappings for Hindi and Urdu alphabets and their vowels are given in Table 2 – 5. In Hindi, SHA (श) and SSA (ष) both represent the sound [ʃ] and have one equivalent symbol in Urdu, *i.e.* Sheen (ش). To make distinction between SHA (श) and SSA (ष) in UIT, they are mapped on S and S1 respectively. Similarly in Urdu, Seh (ث), Seen (س) and Sad (ص) represent the sound [s] and have one equivalent symbol in Hindi, *i.e.* SA (स). To make distinction among them in UIT, they are mapped on s1, s and s2 respectively. All similar cases are shown in Table 6.

IPA	Urdu (UIT)	Hindi (UIT)
ʈ̤	ت (t_d), ط (t_d1), ۃ (t_d2)	त (t_d)
s	ث (s1), س (s), ص (s2)	स (s)
H	ح (h1), ہ (h)	ह (h)

z	ذ (z1), ز (z), ژ (Z), ض (z2), ظ (z3)	ज़ (z)
ʃ	ش (S)	श (S), ष (S1)
r	ر (r)	र (r), ऋ (r1)

Table 6: Multiple Characters for one IPA

Multi-equivalences are problematic for Hindi-Urdu transliteration.

UIT is extendable to other languages like English, French, Kashmiri, Punjabi, Sindhi, *etc.* For example, Punjabi has one extra character than Urdu *i.e.* Rnoon [ɳ] (ڻ), it is mapped on 'n`' in UIT. Similarly, UIT, a phonetic encoding scheme, can be extended to other languages.

All these mappings can be implemented by simple finite-state transducers using XEROX's XFST (Beesley and Karttunen, 2003) language. A sample XFST code is given in Figure 1.

```
read regex [ب -> b, پ -> p, ج -> [d "_" Z] ];
read regex [[ج ھ] -> [d "_" Z "_" h]];
read regex [و -> v, ی -> j  || .#. _ ];
read regex [و -> v, ی -> j || _ [ٰ | ا]];
read regex [ی ->  e || CONSONANTS _ ];
read regex [ ی ->  i  || _ [ٗ | .#.]];
…
read regex [ब -> b, प -> p, ज़ -> z, झ -> [d "_" Z "_" h]];
read regex [अ -> "@", आ -> A, ई -> i || .#. _ ]
…
```

Figure 1: Sample XFST code

Finite-state transducers are robust and time and space efficient (Mohri, 1997). They are a logical choice for Hindi-Urdu transliteration via UIT as this problem could also be seen as string matching and producing an analysis string as an output like finite-state morphological analysis.

5.2 Contextual HUMT Rules

UIT mappings need to be accompanied by necessary contextual HUMT rules for correct Hindi to Urdu transliteration and *vice versa*.

For example, Vav (و) and Choti Yeh (ی) are used to represent vowels like [o], [ɔ], [i], [e], *etc.* but they are also used as consonants. Vav (و) and Choti Yeh (ی) are consonants when they come at the beginning of a word or when they are followed by Alef mada (ٓ) or Alef (ا). Also, Choti Yeh (ی) represents the vowel [e] when it is preceded by a consonant but when it comes at the end of a word and is preceded by a consonant then it represents the vowel [i]. These rules are shown in red colour in Figure 1.

Thus HUMT contextual rules are necessary for Hindi-Urdu transliteration and they can also be implemented as finite-state transducer using XFST. All these rules can't be given here due to shortage of space.

6 HUMT System

The HUMT system exploits the simplicity, robustness, power and time and space efficiency of finite-state transducers. Exactly the same transducer that encodes a Hindi or Urdu text into UIT can be used in the reverse direction to generate Hindi or Urdu text from the UIT encoded text. This two-way power of the finite-state transducer (Mohri, 1997) has significantly reduced the amount of efforts to build the HUMT system. Another very important and powerful strength of finite-state transducers, they can be composed together to build a single transducer that can perform the same task that could be done with help of two or more transducers when applied sequentially (Mohri, 1997), not only allows us to build a direct Hindi ↔ Urdu transducer, but also helps to divide difficult and complex problems into simple ones, and has indeed simplified the process of building the HUMT system. A direct Hindi ↔ Urdu transducer can be used in applications where UIT encoding is not necessary like Hindi-Urdu MT system.

The HUMT system can be extended to perform transliteration between two or more different scripts used for the same languages like Kashmiri, Kazakh, Malay, Punjabi, Sindhi, *etc.* or between language pairs like English–Hindi, English–Urdu, English–French, *etc.* by just introducing the respective transducers in the `Finite-state Transducer Manager` of the HUMT system to build a *multilingual machine transliteration system*.

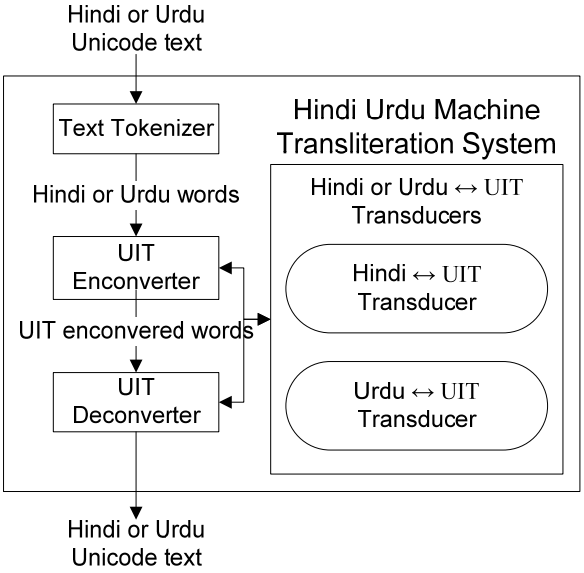

Figure 2: HUMT System

In the HUMT system, `Text Tokenizer` takes the input Hindi or Urdu Unicode text, tokenizes it into Hindi or Urdu words and passes

them to `UIT Enconverter`. The enconverter enconverts Hindi or Urdu words into UIT words using the appropriate transducer from `Finite-state Transducers Manager`, *e.g.* for Hindi words, it uses the Hindi ↔ UIT transducer. It passes these UIT encoded words to `UIT Deconverter`, which deconverts them into Hindi or Urdu words using the appropriate transducer from `Finite-state Transducers Manager` in reverse and generates the target Hindi or Urdu text.

6.1 Enconversion of Hindi-Urdu to UIT

Hindi ↔ UIT transducer is a composition of the mapping rules transducers and the contextual rules transducers. This is clearly shown in figure 3 with a sample XFST code.

```
clear stack
set char-encoding UTF-8
define CONSONANTS [क | ख | ग | घ | ङ | छ | ज];
read regex [ ् -> J, ः -> h, ं -> 0];
read regex [क -> k, ख -> [k "_" h], ग -> g, घ -> [g "_" h], ङ -> [n "@" g], च -> [t "_" S], छ -> [t "_" S "_" h]];
read regex [[क ् क] -> [k k], [क ् ख] -> [k k "_" h], [ग ् ग] -> [g g], [ग ् घ] -> [g g "_" h]];
…
read regex [[क ि] -> [k h], [न] -> [n A], [य ै] -> [j h], [व ै] -> [v h] || .#. _ .#.];
compose net
```

Figure 3: Sample code for Hindi ↔ UIT Transducer

How the HUMT system works is shown with the help of an example. Take the Hindi sentence:

फ़ाख़ता मुहबत और अमन का निशान है

[fɑxəṯɑ mʊhəbəṯ ɔr əmən kɑ nɪʃɑn hæ]

(Dove is symbol of love and peace)

This sentence is received by the `Text Tokenizer` and is tokenized into Hindi words, which are enconverted into UIT words using the mapping and the contextual rules of Hindi ↔ UIT transducer by the `UIT Enconverter`. The Hindi Words and the UIT enconversions are given in Table 7.

Hindi Words	UIT
फ़ाख़ता [fɑxəṯɑ]	fAx@t_dA
मुहबत [mʊhəbəṯ]	mUh@b@t_d
और [ɔr]	Or
अमन [əmən]	@m@n
का [kɑ]	kA
निशान [nɪʃɑn]	nISAn
है [hæ]	H{

Table 7: Hindi Words with UIT

6.2 Deconversion of UIT to Hindi-Urdu

For the deconversion, Hindi ↔ UIT or Urdu ↔ UIT transducer is applied in reverse on the UIT enconverted words to generate Hindi or Urdu words. To continue with the example in the previous section, the UIT words are deconverted into the Urdu words by the `UIT Deconverter` using Urdu ↔ UIT transducer in reverse. The Urdu words are given in table 8 with the Hindi and the UIT words.

Hindi	UIT	Urdu
फ़ाख़ता [fɑxəṯɑ]	fAx@t_dA	فاختا
मुहबत [mʊhəbəṯ]	mUh@b@t_d	مُحبت
और [ɔr]	Or	اور
अमन [əmən]	@m@n	امن
का [kɑ]	kA	کا
निशान [nɪʃɑn]	nISAn	نِشان
है [hæ]	H{	ہَے

Table 8: Hindi, UIT and Urdu Words

Finally, the following Urdu sentence is generated from Urdu words.

فاختا مُحبت اور امن کا نِشان ہَے

Here the word फ़ाख़ता [fɑxəṯɑ] (Dove) is transliterated wrongly into 'فاختا' because the vowel [ɑ] at the end of some Urdu words (borrowed from Persian language) is transcribed with help of Heh-gol [h] (ہ). This phenomenon is a problem for Hindi to Urdu transliteration but not for Urdu to Hindi transliteration.

7 Evaluation Experiments and Results

For evaluation purpose, we used a Hindi corpus, containing 374,150 words, and an Urdu corpus with 38,099 words. The Hindi corpus is extracted from the Hindi WordNet[2] developed by the Resource Center for Indian Language Technology Solutions, CSE Department, Indian Institute of Technology (IIT) Bombay, India and from the project CIFLI (GETALP-LIG[3], University Joseph Fourier), a project for building resources and tools for network-based "linguistic survival" communication between French, English and Indian languages like Hindi, Tamil, etc. The Urdu corpus was developed manually from a book titled "ظلمت کدہ" [zʊlməṯ kəɖɑ]. The Hindi-Urdu corpus contains in total 412,249 words.

The HUMT system is an initial step to build Urdu resources and add Urdu to the languages of

[2] http://www.cfilt.iitb.ac.in
[3] http://www.liglab.fr

SurviTra-CIFLI (Survival Translation) (Boitet et al, 2007), a multilingual digital phrase-book to help tourists for communication and enquiries like restaurant, hotel reservation, flight enquiry, etc.

To reduce evaluation and testing efforts, unique words are extracted from the Hindi-Urdu corpus and are transliterated using the HUMT system. These unique words and their transliterations are checked for accuracy with the help of dictionaries (Platts, 1911; Feroz).

7.1 Urdu → Hindi Transliteration Results

While transliterating Urdu into Hindi, multiple problems occur like multi-equivalences, no equivalence, missing diacritical marks in Urdu text. For example, Sheen [ʃ] (ش) can be transliterated in Hindi into SHA [ʃ] (श) or SSA [ʃ] (ष) that are present in 7,917 and 6,399 corpus words respectively. Sheen [ʃ] (ش) is transliterated into SHA [ʃ] (श) by default. Thus, 6,399 words containing SSA [ʃ] (ष) are wrongly transliterated into Hindi using HUMT. Urdu to Hindi multi-equivalences cases are given in Table 9 with their frequencies.

Urdu	Hindi (corpus Frequency)
ش [ʃ]	श (7917), ष (6399)
ر [r]	र (79,345), ऋ (199)

Table 9: Urdu → Hindi Multi-equivalences

Some Hindi characters do not have equivalent characters in Urdu, *e.g.* NNA [ɳ] (ण), retroflexed version of [n], has approximately mapped onto Noon [n] (ن). This creates a problem when a word actually containing NNA [ɳ] (ण) is transliterated from Urdu to Hindi. No-equivalence cases are given in Table 10.

Urdu	Hindi (corpus Frequency)
-	ण (4744)
-	ङ (0)
-	ञ (532)

Table 10: Urdu → Hindi No-equivalences

Missing diacritical marks is the major problem when transliterating Urdu into Hindi. The importance of diacritical marks has already been explained in section 4.3. This work assumed that all necessary diacritical marks are present in Urdu text because they play a vital role in Urdu to Hindi transliterations. Results of Urdu to Hindi transliteration are given in Table 11.

	Error Words	Accuracy
Corpus	11,874	97.12%
Unique Words	123	98.54%

Table 11: Urdu → Hindi Transliteration Results

7.2 Hindi → Urdu Transliteration Results

Hindi → Urdu transliteration also have multi-equivalences and no-equivalence problems that are given in Table 12.

Hindi	Urdu (corpus Frequency)
त	ت (41,751), ط (1312)
स	س (53,289), ص (751), ث (86)
ह	ہ (72,850), ح (1800)
ज़	ز (2551), ض (1489), ذ (228), ظ (215), ژ (2)
-	ع (2857)

Table 12: Hindi → Urdu Multi & No equivalences

Results of Hindi to Urdu transliteration are given in Table 13.

	Error Words	Accuracy
Corpus	8,740	97.88%
Unique Words	1400	83.41%

Table 13: Hindi → Urdu Transliteration Results

Interestingly, Hindi to Urdu conversion is 14.47% less accurate on the unique words as compared to its result on the corpus data that is a contrasting fact for the reverse conversion.

The HUMT system gives 97.12% accuracy for Urdu to Hindi and 97.88% accuracy for Hindi to Urdu. Thus, the HUMT system works with 97.50% accuracy.

8 Future Implications

Hindi-Urdu transliteration is one of the cases where one language is written in two or more mutually incomprehensible scripts like Kazakh, Kashmiri, Malay, Punjabi, Sindhi, *etc.* The HUMT system can be enhanced by extending UIT and introducing the respective finite-state transducers. It can similarly be enhanced to transliterate between language pairs, *e.g.* English-Arabic, English-Hindi, English-Urdu, French-Hindi, *etc.* Thus, it can be enhanced to build a *multilingual machine transliteration system* that can be used for *cross-scriptural transliteration* and MT.

We are intended to resolve the problems of multi-equivalences, no-equivalences and the most importantly the restoration of diacritical marks in Urdu text that are observed but left unattended in the current work. Restoration of diacritical marks in Urdu, Sindhi, Punjabi, Kashmiri, *etc.* texts is essential for word sense disambiguation, natural language processing and speech synthesis of the said languages.

The HUMT system will also provide a basis for the development of *Inter-dialectal* translation system and MT system for *surface-close* languages like Indonesian-Malay, Japanese-Korean,

Hindi-Marathi, Hindi-Urdu, *etc.* Translation of the *surface-close* languages or *inter-dialectal* translation can be performed by using mainly transliteration and some lexical translations. Thus HUMT will also provide basis for *Cross-Scriptural Transliteration, Cross-scriptural Information Retrieval*, *Cross-scriptural Application Development*, *inter-dialectal translation* and *translation of surface-close languages*.

9 Conclusion

Finite-state transducers are very efficient, robust, and simple to use. Their simplicity and powerful features are exploited in the HUMT model to perform Hindi-Urdu transliteration using UIT that is a generic and flexible encoding scheme to uniquely encode natural languages into ASCII. The HUMT system gives 97.50% accuracy when it is applied on the Hindi-Urdu corpora containing 412,249 words in total. It is an endeavor to bridge the scriptural, ethnical, cultural and geographical division between 1,017 millions people around the globe.

Acknowledgement

This study is partially supported by the project CIFLI funded under ARCUS-INDIA program by *Ministry of Foreign Affairs* and *Rhône-Alpes* region.

References

Beesley, Kenneth R. and Karttunen, Lauri. 2003. *Finite State Morphology*. CSLI Publications, USA.

Boitet, Christian. Bhattacharayya, Pushpak. Blanc, Etienne. Meena, Sanjay. Boudhh, Sangharsh. Fafiotte, Georges. Falaise, Achille. Vacchani, Vishal. 2007. *Building Hindi-French-English-UNL Resources for SurviTra-CIFLI, a linguistic survival system under construction*. Proceedings of the Seventh Symposium on NLP, 13 – 15 December, Chonburi, Thailand.

Feroz ul Din. فیروزُاللغات اُردو Feroz Sons Publishers, Lahore, Pakistan.

Hussain, Sarmad. 2004. *Letter to Sound Rules for Urdu Text to Speech System*. Proceedings of Workshop on Computational Approaches to Arabic Script-based Languages, COLING 2004, Geneva, Switzerland.

James, L. Hieronymus. 1993. *ASCII Phonetic Symbols for the World's Languages: Worldbet*. AT&T Bell Laboratories, Murray Hill, NJ 07974, USA.

Kellogg, Rev. S. H. 1872. *A Grammar of Hindi Language*. Delhi, Oriental Book Reprints.

Khan, Mehboob Alam. 1997. اُردو کا صوتی نظام (Sound System in Urdu) National Language Authority, Pakistan.

Knight, K. and Graehl, J. 1998. *Machine Transliteration.* Computational Linguistics, 24(4).

Knight, K. and Stall, B G. 1998. *Translating Names and Technical Terms in Arabic Tex.* Proceedings of the COLING/ACL Workshop on Computational Approaches to Semitic Languages.

Malik, M. G. Abbas. 2006. *Punjabi Machine Transliteration*. Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, July 2006, Sydney.

Mohri, Mehryar. 1997. *Finite-state Transducers in Language and Speech Processing.* Computational Linguistics, 23(2).

Montaut A. 2004. *A Linguistic Grammar of Hindi*. Studies in Indo-European Linguistics Series, München, Lincom Europa.

Paola, V. and Sanjeev, K. 2003. *Transliteration of proper names in cross-language applications.* Proceedings of the 26th annual International ACM SIGIR conference on research and development in information retrieval.

Pirkola, A. Toivonen, J. Keskustalo, H. Visala, K. and Järvelin, K. 2003. *Fuzzy translation of cross-lingual spelling variants*. Proceedings of the 26th Annual international ACM SIGIR Conference on Research and Development in informaion Retrieval, Toronto, Canada.

Platts, John T. 1909. *A Grammar of the Hindustani or Urdu Language*. Crosby Lockwood and Son, 7 Stationers Hall Court, Ludgate hill, London. E.C.

Platts, John T. 1911. *A Dictionary of Urdu, Classical Hindi and English*. Crosby Lockwood and Son, 7 Stationers Hall Court, Ludgate hill, London, E.C.

Rahman, Tariq. 2004. *Language Policy and Localization in Pakistan: Proposal for a Paradigmatic Shift*. Crossing the Digital Divide, SCALLA Conference on Computational Linguistics.

Rai, Alok. 2000. *Hindi Nationalism*. Orient Longman Private Limited, New Delhi.

Wells, J C. 1995. *Computer-coding the IPA: A Proposed Extension of SAMPA*. University College London. http://www.phon.ucl.ac.uk/home/sampa/ipasam-x.pdf.

Yan Qu, Gregory Grefenstette, David A. Evans. 2003. *Automatic transliteration for Japanese-to-English text retrieval*. Proceedings of the 26th annual interntional ACM SIGIR conference on Research and development in information retrieval.

Zia, Khaver. 1999a. *Standard Code Table for Urdu*. Proceedings of 4th Symposium on Multilingual Information Processing (MLIT-4), Yangon, Myanmar, CICC, Japan.