Roman to Urdu Transliteration using word list

Tafseer Ahmed
Universitaet Konstanz
tafseer@gmail.com

Abstract

The paper discusses transliteration of Urdu words from roman script to Urdu script. We propose a word list based approach that gives a better transliteration to the Persio-Arabic letters that have same or similar sound but are written as different letter in Urdu script. We give a rule based system for roman to Urdu transliteration. As the roman script for Urdu does not follow any standard and a single word can be written in multiple ways, the proposed rule based system covers the different ways of writing and give the best possible Urdu transliteration based on word list and roman to Urdu script mapping rules.

1. Introduction

Urdu is an Indo-Aryan language that uses extended Persio-Arabic script. As Urdu has many sounds that are not present in Persian and Arabic, the Persio-Arabic script is modified to cover Urdu text too. An example of extension is the Urdu character “ھ” (*do chasmi hay)* that represents the aspiration. Similarly, a symbol for retroflex “ڈ” is introduced in Urdu by extending the symbol of dental “د”.

Urdu has borrowed significant number of words from Arabic and Persian glossary. The borrowed words are written faithfully in Urdu as those are written in its original language. But Urdu speakers do not pronounce the words in the same way as they are pronounced by an Arab or Persian person. For example, an Arabic speaker can differentiate between Alif “ا” and Ain “ع”, but for an Urdu speaker both have the same sound. As we will see in next section, it causes many problems in roman to English transliteration.

The roman script is not an official standard for writing Urdu text, but it is widely used. The reason is influence of English language in Urdu speaking community. English is widely used in offices, business circle and education sector. English is also the default language of user interfaces of computers. Any person can install Urdu support on the computer and can write email and use chat applications in Urdu, but roman script is widely used when there comes a need of a language for the informal communication over internet.

Unlike Urdu script, the roman script for Urdu does not have any standard for spelling the words. A word can be written in various forms not only by distinct writers but also by the same writer at different occasions. Specially, there is no one to one mapping between Urdu letters for vowel sounds and the corresponding roman letters. The support of Urdu script on the computers is getting better by the usage of Unicode character set and Open Type fonts. The availability of Urdu script support demands for roman to Urdu transliterator that can convert the text written in roman Urdu into proper Urdu script.

This paper focuses on the issues related to the transliteration of Urdu text from roman script to Urdu script. In the paper we have analyzed roman Urdu data and identified different spelling rules of roman Urdu. We give an algorithm that can transliterate roman word written by using different spellings to Urdu script using a word list.

2. Issues in Roman-Urdu Transliteration

We do not find any detailed analysis of automatic transliteration of Urdu text in Roman script to Urdu script. The mapping of English and Urdu characters is discussed earlier [1]. Transliteration of English words (written in roman script) into Urdu script is also discussed [2].

The roman script for Urdu does not follow any standard. In this study, we analyzed roman text available at different websites and jotted down the spelling rules and issues that are commonly found in roman Urdu script.

Roman to Urdu transliteration cannot be implemented by simple one to one replacement of characters. There are many problems in this regard. The following discussion lists the issues that create problem in roman to Urdu transliteration.

2.1. Multiple Urdu letters for roman consonants

Roman script does not have characters for all consonants that are used in Urdu script. An Urdu speaker uses a single roman character for more than one Urdu characters. Both عام ‘common’ and آم ‘mango’ are written as *aam* in roman script. The same sound Persio-Arabic characters are not the only problem in roman to Urdu mapping. Different letters (or letter sequences) for different Urdu consonant can map to single roman equivalent. Urdu letter(s) “چ” (*chay)* and “چه” (*chay-do_chasmi_hay)* both are written as *ch* in roman script as in *chor* چور ‘thief’ and *churi* چهری ‘knife’.

Table 1 gives the list of same sound Urdu characters and corresponding roman characters. The first column of the table has a roman character or character sequence. The second column has all the Urdu letters that are used as equivalent of the roman character in the previous column. The last column has the most frequently used Urdu character against each roman character(s). These characters can be used as the default transliteration for the roman character(s) specified in front of the each equivalent symbol.

Table 1: List of roman characters that has multiple equivalent characters

Roman letter(s)	**Equivalent Urdu letters**	**Most common Urdu equivalent**
a	“ا” (alif), “ع” (ain), “ء” (hamza), “آ” (alif_mad)	“ا” (alif)
ch	“چ” (chay), “چه” (chay-do_chasmi_hay)	“چ” (chay)
d	“د” (daal) , “ڈ” (ddaal)	“د” (daal)
gh	“غ” (ghain), “گـــه” (gaaf-dochasmi)	“غ” (ghain)
h	“ح” (hay), “ه” (hay_gol), “ھ” (hay_dochasmi)	“ه” (hay_gol)
kh	“خ” (khay) , “کـــه” (kaf-dochasmi)	“خ” (khay)
K	“ک” (kaaf), “ق” (qaaf)	“ک” (kaaf)
R	“ر” (ray), “ڑ” (rray)	“ر” ray
s	“ث” (say), “س” (seen), “ص” (suad)	“س” (seen)
T	“ت” (tay), “ط” (tuay), “ٹ” (ttay)	“ت” (tay)
y	“ی” (chooti-ye), “ء” (hamza)	“ی” (chooti-ye)
z	“ذ” (zaal), “ز” (zay), “ض” (zuaad), “ظ” (zuay), “ژ” (jay)	“ز” (zay)
N	“ن” (noon), “ں” (noon-ghunna)	“ن” (noon)

The above table shows that the roman letter ‘h’ marks Urdu “ح” (*hay)*, “ه” (*gol_hay)* and “ھ” (*do chasmi hay)*. When it comes after an aspiratable consonants like ‘b’, ‘p’ or ‘k’, it represents aspiration i.e. *do chashmi hay*. The Urdu letter “ق” (*qaf)* is written as Roman ‘q’. Occasionally, it can be written as ‘k’ according to the choice of the writer. Similarly, frequently used mapping of all roman characters to Urdu characters is listed in the above table.

2.2. Multiple roman letters for Urdu vowels

Representation of vowels in roman Urdu is a complex issue. In roman Urdu, the same character can be used for both short and long vowel. For example, ‘a’ is used in *jang* جنـگ ‘war’ and *bang* بانـــگ ‘call’ as short and long vowel respectively. An Urdu vowel can be represented by more than one roman letters/ letter sequences. For example, بانـــگ can also be written as ‘baang’. Hence long ‘a' sound can be expressed either by ‘a’ or ‘aa’.

Like many other issues, roman Urdu inherits this problem from roman orthography of English. In English script, a roman vowel character can map to different vowel sounds e.g. ‘a’ maps to short vowel in the English word ‘about’ and to long vowel in the word ‘father’.

Table 2 gives the list of roman characters used for vowel sounds and the corresponding Urdu characters.

Table 2: Roman letters for Urdu Vowels

Urdu letter	**Roman Equivalents**
Zabar	a
Zer	i
Paish	u
alif ا	a, aa
bari-ye ے	ai , ay, ei, e
chooti ye ی	ee, ey, i , ie
vao و	oo, au, ou, o, u

The first three vowels (written in italics) in the above table are short vowels and are not written in Urdu script.

2.3. Y behaves as consonant as well as vowel

The roman letter ‘y’ represents Urdu consonant *ye*. In Urdu word *yaqeen* يقين ‘belief’, ‘y’ is used as a consonant. But ‘y’ is part of vowel sequences too, as can be seen in table 2. For example *hay* ہے ‘is’ has roman letter sequence ‘ay’ for Urdu “ے” (*bari ye).* This implies that it is ambiguous during transliteration whether this particular instance of ‘y’ is a consonant or part of the vowel sequence. As ‘y’ is part of only two vowel sequences (‘ay’ and ‘ey’), it acts as a vowel only if it follows ‘a’ or ‘e’. In all the other contexts, it acts as a consonant.

The consonant ‘y’ does not always map on *ye*. It maps on “ء” (*hamza)* in certain contexts. When followed by ‘i’, ‘e’, ‘o’, it can represent Urdu *hamza* too as in *gayi* گئی ‘(she) went’, *gaye* گئے ‘(they) went’ and *jayo* جائو ‘(you) go’. Hence the consonant ‘y’ that precedes ‘i’, ‘o’ and ‘u’ can map on both Urdu “ی” (*ye)* and “ء” (*hamza).*

2.4. Double roman letters for germination

In Urdu script, “ّ” (*tashdeed*) is used as germination mark. This marker written on (after) a letter means that this letter will be repeated. The Urdu word الو ‘owl’ has a single “ل” (*laam)* but it is repeated because it has a diacritical tashdeed sign on it. In its roman Urdu transcription ‘ullo’, we use double ‘l’. Similarly, in the word بدھو ‘stupid’, the consonant “د” (*daal)* (or “دھ” *daal – do chasmi hay*) has tashdeed sign. Its roman equivalent has double ‘d’ as ‘buddho’. The transliterator should treat double consonants in roman script as single consonant in Urdu script. As diacritic marks are mostly not used in writing, “ّ” (*tashdeed*) too is not considered while writting Urdu text.

2.5. Vowel change around gol hay

Another problem in roman to Urdu transliteration is the short vowel around “ہ” (*gol_hay).* The word شہر has (unwritten) short vowel “َ” (*zabar)* between “ش” and “ہ”, and “ہ” and “ر”. In pronunciation, the word has vowel ‘e’ (in place of short vowel ‘a’) and it is written in roman Urdu as ‘*sheher’*. Similarly Urdu word شہرت ‘fame’ has short vowel “ُ” (*pesh)* after “ش” .It is written in roman Urdu as ‘*shohrat’* (and not ‘*shuhrat*’).

2.6. Vowel at start of the word/syllable

In Urdu script, a vowel cannot occur at the start of a syllable without having an “ا” (*alif),* “ع” (*ain)* or “ء” (*hamza)* before it. *Alif* or *Ain* occurs at the start of the word. For example, both the roman words ‘alag’ and ‘adal’ have two (short vowel) a’s which are equivalent to Urdu diacritic mark “َ” (*zabar).* In the Urdu script, the second (in the middle of the word) short vowel ‘a’ will be written as *zabar*, but the first *zabar* must be preceded by a consonant. The original Urdu transcription of these words is الگ ‘separate’ and عدل ‘justice’ respectively.

Urdu letter “آ” (*alif-mad)* is equivalent to two “ا” (*alif-s).* The long vowel ‘aa’ at the start of a roman word is equivalent to either *alif-mad* or *ain - alif* as both آم and عام are written as ‘aam’ in roman Urdu.

The rule for the vowel at the start is not applied on ‘a’ vowel sound only. Any vowel needs an ‘a’ (*alif, hamza* or *ain*) consonant before it. The roman word ‘aur’ have vowel sequence ‘au’ at its start. Table 2 tells that it is equivalent to Urdu “و” (*vao).* In Urdu the word is written as اور with an *alif* before the *vao*.

The same phenomenon occurs at the start of other syllables. If a vowel occurs at the start of a non-word-initial syllable, it must be preceded by either Urdu “ع” (*ain)* or “ء” (*hamza).* The Urdu word for roman lexeme “bha.i” is بھائی ‘brother’. The second syllable of this roman word is the single letter ‘i’, but Urdu script needs a *hamza* before it. Similarly, roman word ‘sa.ee’ ‘struggle’ is equivalent to Urdu سعی with *ain* at the start of second syllable.

2.7. Short vowel at the end of word

Urdu script does not allow short vowel at the end of the word. We know that roman vowel ‘i’ at initial and middle position (in the word) can map either on short-i “ِ” (*zer*) or long-i “ی” (*chooti-ye*). But ‘i’ at the end of the word will always map on the long vowel i.e. “ی” (*chooti-ye).* It is same for other short vowels too.

2.8. bari-ye at end of the word

When “ی” (*chooti-ye)* and “ے” (*bari-ye)* are used as vowels, they represent different sounds. In Urdu script, *chooti-ye* is used for both *chooti-ye* and *bari-ye* at the middle of the word. For example, Urdu word “ہیں” has *chooti-ye* for ‘ai’ sound. But at the end of the word, the same vowel sound is represented by *bari-ye*. For example, “ہے” has the same vowel but now mapped on *bari-ye*.

2.9. Syllable Boundaries

We cannot predict the syllable boundary either in roman script or in Urdu script. But for transliteration, we need to know about the boundary in roman words. If the vowel appears after the consonant in the syllable, it will be transliterated by using the general rules. But if it appears at the start of the syllable, special rules as discussed in 2.6 will be applied on it.

2.10. Roman vowels in sequence

Roman vowels when appears in the sequence are considered as the special case of syllable boundary problem. Table 2 shows that a single Urdu vowel can map either on one letter or on two letter sequence in roman script. When a valid two letter sequence appears in a roman word, it can either be taken as equivalent of one Urdu vowel or two vowels.

Take the example of roman letter sequence ‘ai’ in *bhai* بھـــائ ‘brother’. In this word ‘a’ stands for Urdu “ا” *(alif)* and ‘i’ stands for *hamza – chooti ye* ئ. But the same sequence represents single letter (Urdu *ye*) in *main* میں ‘I’.

3. Algorithm for Roman to Urdu transliteration

In the above section, we discussed about the issues in roman to Urdu transliteration. In this section, we will discuss the methods to solve these issues and will propose an algorithm for the transliteration.

3.1. Word List Approach

We argue that the transliteration can give better results if we use a word list along with the mapping rules. As we have seen in section 2, there is no one to one mapping between the two scripts. A roman letter or letter sequence can map on more than one letters in Urdu script. If we consult a word list, we can find the relevant mapping in the given context.

An important issue in roman to Urdu transliteration systems is same sound characters. Another area which is most effected by the same problem is “spell checking”. One can make a mistake of writing ‘d’ in place of ‘t’ or vice versa in English. A solution of this problem is soundex algorithm [3]. The same problem occurs in Urdu spell checking. Naseem has introduced a soundex algorithm for Urdu [4].

In soundex algorithm, the words are encoded in such a way that the common sound characters get the same code. For example, “ذ” (zaal), “ز” (zay), “ض” (zuaad) and “ظ” (zuay) have same code in Urdu [4].

To find the appropriate Urdu word for a roman transcription, we can encode words in roman and Urdu script in such a way that the encoded results of both can be compared. The available Urdu encoding scheme will not work in the transliteration problem because, transliteration involves few other issues too; For example, roman ‘ch’ maps on both “چ” *(chay)*, “چھ” (chay-*do_chasmi_hay)*. This pair does not appear in Naseem’s list. So we need a different encoding scheme for the purpose of roman to Urdu transliteration.

Moreover, we have discussed the issues related to vowel mapping. These issues can not be handled by a simple soundex algorithm. We need a new algorithm that uses character classes like soundex algorithm but can deal the above described issues too.

Another relevant algorithm is the Urdu to Hindi transliteration algorithm given by Abbas Malik [5]. As Urdu script does not have diacritic marks, he generated all possible combination of the word (with unwritten diactrics) for Urdu to Hindi transliteration. For example, for the word بلی ‘cat’, the generated words will be: ‘bilii’, ‘balii’, ‘bulii’, ‘billi’, ‘balli’, ‘bulli’. Out of these possibilities, only one is present in Hindi word list and is selected as the correct input.

If there is one to many mapping between a roman letter or a letter sequence and its Urdu equivalents, we can generate multiple words and match those with its Urdu equivalent. But the discussion in section 2 suggests that generating all possible combinations of Urdu (script) words from a roman (script) word is not a single step method. There are conflicting mapping rules that can be applied on a vowel or vowel sequence. We need an algorithm that applies the rules in some order and deals all the issues listed in the above section. So we may use the fundamental concept of Malik’s algorithm, but we need to enhance it to deal with the multiple interpretations of a sequence of the characters.

3.2. Encoding Scheme for transliteration

As described above, we propose an algorithm in which Urdu words are encoded in a similar way as in soundex algorithm. The roman encoded words will be compared with these Urdu encoded words and the consonants of roman Urdu words will be encoded by a similar scheme. The vowels of roman word need special processing. We will generate all possible (encoded) outputs related to the vowel sequences and will then match those with encoded Urdu words.

Table 3 gives the encoding scheme for Urdu words. The list has all the (non-diacritic) characters of Urdu.

Table 3: Encoding for Urdu charcaters

Code	Corresponding Urdu letter(s)	Default Urdu letter
A	“ا” (alif), “ع” (ain)	“ا” (alif)
AA	“آ” (alif_mad)	“آ” (alif_mad)
B	“ب” (bay)	“ب” (bay)
P	“پ” (pay)	“پ” (pay)
T	“ت” (tay), “ط” (tuay), “ٹ” (ttay)	“ت” (tay)
J	“ج” (jeem)	“ج” (jeem)
S	“ث” say, “س” (seen), “ص” (suad)	“س” (seen)
CH	“چ” (chay)	“چ” (chay)
H	“ح” (hay), “ہ” (hay_gol), “ھ” (hay_dochasmi)	“ہ” (hay_gol)
KH	“خ” (khay)	“خ” (khay)
D	“د” (daal) , “ڈ” (ddaal)	“د” (daal)
Z	“ذ” (zaal), “ز” (zay), “ض” (zuaad), “ظ” (zuay), “ژ” (jay)	“ذ” (zaal)
R	“ر” (ray), “ڑ” (rray)	“ر” (ray)
SH	“ش” (sheen)	“ش” (sheen)
GH	“غ” (ghain)	“غ” (ghain)
F	“ف” (fay)	“ف” (fay)
K	“ک” (kaaf), “ق” (qaaf)	“ک” (kaaf)
G	“گ” (gaaf)	“گ” (gaaf)
L	“ل” (Laam)	“ل” (Laam)
M	“م” (meem)	“م” (meem)
N	“ن” (noon), “ں” (noon-ghunna)	“ن” (noon)
O	“و” (wao), hamza-wao	“و” (wao)
Y	“ی” (chooti-ye), “ء” (hamza)	“ی” (chooti-ye)
E	“ے” (bari-ye)	“ے” (bari-ye)

Now we will illustrate encoding examples of few Urdu words. The word بخار ‘fever’ is encoded as ‘*BKHAR*’. شکار ‘hunting’ is encoded as ‘*SHKAR*’. دانت ‘tooth’ is encoded as ‘*DANT*’ and ڈھول ‘drum’ is encoded as ‘*DHOL*’.

3.3. Transliteration Algorithm

The transliteration process consists two parts. In the first part, a list of most frequently used Urdu words is encoded by using the encoding scheme given in table 3. For this purpose a list of 5000 frequently used Urdu words with frequencies can be used, which is available at [6]. We can encode each word of this list using the above scheme. The resulting list contains the encoded word, the real Urdu word and its frequency.

The second part is comprised of the transliteration of roman words. The following algorithm encodes a roman Urdu word and matches it with the list of frequently used Urdu words. The algorithm accepts the roman word (*rom_word)* as input.

After each step of the algorithm, a brief explanation is given. We selected following roman words ‘*alag*’, ‘*ullo*’, ‘*bukhar*’, ‘*bhai*’, ‘*hai*’, ‘*bhayi*’ and ‘*shohrat*’ as example. The step by step processing on these words is shown.

The encoding of intended Urdu words of these roman words are: الگ ALG, الو ULO, بخار BKHAR, بھائ BHAYI, ہے HE, بھائ BHAYI, شہرت SHHRT

1) Except ‘a’, ‘e’, ‘i’, ‘o’, ‘u’, ‘y’ and ‘h’, change the case of all the characters of *rom_word* into capital. This transformed encoded word is termed as *enc_rom_word*.

(**Explanation:** As vowel mapping need more complex processing than one to one replacement, the encoding is applied on consonants only.

A character in capital case means a rule is already applied on it and the following low pirority rule will not be accidently applied on it.

Example Words : ‘aLaG’, ‘uLLo’, ‘BuKhaR’, ‘Bhai’, ‘hai’, ‘Bhayi’, ‘ShohRaT’)

2) If the two consequent capital letters are the same, delete one of those double letters.

(**Explanation:** The rule deals the germination/tashdeed as explained in 2.4. it deletes one of the double consonants because the germinated consonant is written only once in Urdu script.

Example Words: ‘aLaG’, ‘uLo’, ‘BuKhaR’, ‘Bhai’, ‘hai’, ‘Bhayi’, ‘ShohRaT’)

3) If the word begins with a vowel (any roman letter or letter sequence present in Table 2), append ‘A’ at the beginning of the word.

(**Explanation:** As discussed in 2.6, we need an extra ‘a’ sound character before a vowel at the start of the word in Urdu script. For this purpose, we introduce ‘A’ in *enc_rom_word* that will get matched with Urdu equivalent. For example, Urdu word اینٹ ‘brick’ is written as ‘*eent*’ in roman script. In the roman word, ‘ee’ stands for Urdu “ی” but we need to put a “ا” at the start.

Example Words : ‘AaLaG’, ‘AuLo’, ‘BuKhaR’, ‘Bhai’, ‘hai’, ‘Bhayi’, ‘ShohRaT’)

4) For the sequences ‘eh’ and ‘oh’, do the following replacements. Consider the longest match at left hand side.

ehe = eHe , H

eh = eH , H
oh = oH , H
h = H

(Explanation: In our mapping rules, if there are more than one potential replacements at right hand side corresponding to a single left hand side, then the system makes *n* number of copies of *enc_rom_word*. Each of the possible right hand side replacement is applied on one of the copies, and the subsequent steps of algorithm are applied on each of those.

The above mapping rules deal with the unwritten long vowel before a "ہ" (*gol-hay)* in Urdu script. It is discussed in 2. 5.

Example Words : 'aLaG', 'uLo', 'BuKHaR', 'BHai', 'Hai', 'BHayi', 'SHoHRaT'/'SHHRaT')

5) If 'y' is the last character of the word and is preceded by 'e' or 'a', 'then
ey = Y
ay = E

(**Explanation:** As discussed in 2.8, both *chooti-ye* and *bari-ye* are written as "ی" (*chooti-ye)* 'Y' in medial position, but *bari-ye* is written as "ے" (*bari-ye)* 'E' only at the final position.

Example Words : 'AaLaG', 'AuLo', 'BuKHaR', 'BHai', 'Hai', 'BHayi', 'SHoHRaT'/'SHHRaT')

6) If 'y' is preceded by 'e' or 'a' and followed by a vowel then
ey = Y, eY
ay = Y, aY

(**Explanation:** The 'y' in this case can act either as consonant or as part of a vowel sequence. For example, in *gayi* گئی '(she) went', 'y' acts as consonant. Step 8 gives more details about the rules of this type.

Example Words : 'AaLaG', 'AuLo', 'BuKHaR', 'BHai', 'Hai', 'BHYi'/ 'BHaYi', 'SHoHRaT'/ 'SHHRaT')

7) Change the case of 'y' as capital.
y = Y

(**Explanation:** As we have dealt with all the special cases of 'y', this general rule changes the case of the remaining ones as capital.

Example Words : 'AaLaG', 'AuLo', 'BuKHaR', 'BHai', 'Hai', 'BHYi'/ 'BHaYi', 'SHoHRaT'/ 'SHHRaT')

8) If the vowel sequence 'ai' or 'ei' is present at the end of the word, then apply following replacement.
ai = E, aYi, aAi
ei = E, eYi, eAi

(**Explanation:** As discussed in 2.10, the character sequence 'ai' either correspond to single letter "ے" (*bari-ye)* or it is a sequence of two vowels in two different syllables. In this case, 'a' is in the first syllable and 'i' is the start of the second syllable. As we need "ء" (*hamza)* or "ع" (*ain)* before the vowel at syllable initial position, 'Y' and 'A' are introduced to represent these characters.

Example Words: 'AaLaG', 'AuLo', 'BuKHaR', 'BHE'/'BHaYi' 'HE'/'HaYi'/'HaAi', 'BHYi'/ 'BHaYi'/'BHaAi', 'SHoHRaT'/ 'SHHRaT')

9) If there is a sequence (*seq)* of two or more vowels, then find all the combinations of valid vowel sequences *seq1*, *seq2*, ….., *seqn*, and put 'A' for "ع" (*ain)* or 'Y' for "ء" (*hamza)* is put between these valid sequences.

(**Explanation:** This rule is a generalized form of rule 8. It deals with all possible interpretations of sequence of vowels. An example of two letter sequences is 'ai' that has two valid vowel combinations 'a-i' and 'ai' (as given in table 2). By applying the rule, we get 'aAi', 'aYi' and 'ai' for further processing.

Another two letter sequence is 'ua' that has only one valid combination 'u-a'. The other possibility 'ua' is not a valid vowel combination because 'ua' does not map on any single Urdu vowel. Hence, we get 'uYa' and 'uAa' for further processing in subsequent steps.

An example of three vowel letters in a row is 'aai' آئی '(she) came'. It has three valid sequences 'a-ai', 'aa-i', 'a-a-i'. By applying the rule, we get 'aAai', 'aYai', 'aaAi', 'aaYi', 'aAaAi', 'aAaYi', 'aYaYi' and 'aYaAi' for further processing.

Example Words : 'AaLaG', 'AuLo', 'BuKHaR', 'BHE'/'BHaYi' 'HE'/'HaYi'/'HaAi', 'BHYi'/ 'BHaYi'/'BHaAi', 'SHoHRaT'/ 'SHHRaT')

10) For two vowel sequence, do the following replacements.
aa = A
ai = Y
ei = Y
ee = Y
ie = Y
oo = O
au = O
ou = O

(**Explanation:** It is simple one to one mapping of vowel sequence with encoding of corresponding Urdu letter.

Example Words : 'AaLaG', 'AuLo', 'BuKHaR', 'BHE'/'BHaYi' 'HE'/'HaYi'/'HaAi', 'BHYi'/ 'BHaYi'/'BHaAi', 'SHoHRaT'/ 'SHHRaT')

11) Search the following vowels at word's final position and . make the substitutions accordingly.

e (at final) = E
a (at final) = A, H
i (at final) = Y
u (at final) = O

(**Explanation:** In Urdu script, the word's final vowel is always written as long vowel. For example the final 'i' of *aadmi* 'man' is not ambiguous between short vowel (unwritten diacratic *zer*) and long vowel (*chooti-ye*). Only *chooti-ye* can appear at the end of any word.
The final 'a' can map on "ہ" (gol-hay) too. For example, the final 'a' of roman *sada* سادہ 'simple' stands for gol-hay.

Example Words : 'AaLaG', 'AuLO', 'BuKHaR', 'BHE'/'BHaYY' 'HE'/'HaYY'/'HaAY', 'BHYY'/ 'BHaYY'/'BHaAY', 'SHoHRaT'/ 'SHHRaT')

12) Search for the following vowel sequences and make the following replacements.

a = null, A
i = null, Y
u = null, O
e = E
o = O

(**Explanation:** After dealing the vowels at initial and final positions, and dealing the special cases of vowel sequence, the general rule of vowel sequence replacement is given.

This is the last step for encoding a roman word. The following steps search the equivalent of the encoded word in the encoded list of Urdu words.

Example Words :
'AALAG'/ 'ALAG'/ 'AALG'/ 'ALG',
'AOLO'/ 'ALO',
'BOKHAR'/'BKHAR'/'BOKHR'/'BKHR',
'BHE'/'BHAYY'/'BHYY',
'HE'/'HAYY'/'HYY'/ 'HAAY'/'HAY',
'BHYY'/'BHAYY'/'BHYY'/'BHAAY'/'BHAY',
'SHOHRAT'/'SHOHRT'/ 'SHHRAT'/ 'SHHRT')

13) If only one copy of *enc_rom_word* is in processing, search this encoded word in the list of frequently used Urdu word list.

a. If a single match is found, output the Urdu word corresponding to this encoded word.
b. If a list of matched words is found, output the Urdu word with the highest frequency.
c. If no match is found, replace each character / character sequence of *enc_rom_word* with the default character(s) of column 3 of Table 5. (Start from left and match the longest sequence.)

(**Explanation:** If multiple matches are found, then the most frequent Urdu word will be the output. If no match is found, we try to give transliteration by replacing the encoding with corresponding default Urdu characters. This replacement does not solve similar sound character issue and other issues but gives the best achievable output.)

14) If more than one *enc_rom_word* are in processing, search each of these in the list of frequently used words, and make a list of found-matches corresponding to all instances of *enc_rom_word*.

a. If the list has a single matched encoded word, output the Urdu word corresponding to this encoded word.
b. If the list has more than one matched encoded words, output the Urdu word with the highest frequency.
c. If the list is empty, replace each character / character sequence of the first instance of *enc_rom_word* with the default character(s) of column 3 of Table 5.

(**Explanation:** In the step 13c, we arbitrarily choose first instance of *enc_rom_word* to be transliterated into Urdu script.

Example Words: 'ALG', 'ALO', BKHAR', 'BHAYY', 'HE', 'BHAYY', 'SHHRT'.
These are the encoded words that get matched in most frequently used Urdu word list.)

4. Other Issues involved in roman Urdu transliteration

The algorithm discussed in the above section deals with the transliteration of Urdu words written in roman script to Urdu script. Transliterating the text written in roman script involves other issues too. English words are often used in roman Urdu. The English word are written in their actual english spelling and rules of roman Urdu spelling are not applied on those. For example, the words 'school', 'man' and 'side' should be 'iskool', men' and 'said' in roman-Urdu if the rules discussed above are followed. But, these words are written with their original English spelling.
The transliteration of 'school', 'man' and 'side' using the above algorithm will be سچول, مان, سیدے. To resolve this problem, the English words in the roman-Urdu text need special treatment. Another new trend is the use of 'sms-lingo' conventions in Urdu too. For example 'k' (English letter 'kay') is used for Urdu "کے".

Beyond the usage of English words, there are problems related to Urdu words too. In some cases, two words are written without space in roman script, but these words are written as two separate words in Urdu script. The examples are “تم”, “نے”, “کے”, “لئے” and “ہو گا” which are written either as ‘tum ne’, ‘ke liye’ and ‘ho ga’ or alternatively as ‘tumne’, ‘keliye’ and ‘hoga’. These issues demand for pre-processing before application of the above algorithm.

5. Conclusion

In this study, we identified the issues involved in transliterating Urdu words written in roman script to the original Urdu script. We find that same/similar sound consonants are not the only problem in roman to Urdu transliteration. The transliteration of roman vowel letters is also a complex issue as there is no one to one mapping in roman and Urdu script, and Urdu has special orthographical rules for vowels.

We proposed a word list approach for improved transliteration results. The algorithm first encodes frequently used Urdu words then it encodes the roman word. There exists a probability that it will generate more than one encoded word corresponding to a single roman word. The encoded word(s) corresponding to the roman word is/are searched in the list of encoded Urdu word, and the match with the highest frequency is output as the result.

The proposed algorithm deals all the roman word transliteration issues identified during this study.

6. References

[1] Saleem, Muhammad, “Urdu Rasm-ul-Khat Ki Jamiat”, *Akhabar-e-Urdu,* April-May 2002.

[2] Ahmed, Tafseer, “English to Urdu Transliterator”, *CRULP Annual Student Report*, Lahore, 2004.

[3] Russell, Robert, U.S. patent number no. 1261167, 1918.

[4] Naseem, Tahira, *A Hybrid Approach for Urdu Spell Checking*, MS Thesis, NUCES, Lahore, 2004.

[5] Malik, Abbas, *Hindi Urdu Machine Transliteration System*, Masters Thesis, University of Paris, Paris, 2006.

[6] *Urdu 5000 most Frequently Used Words*, CRULP, Lahore, 2008. Available:http://www.crulp.org/software/ling_resources/UrduHighFreqWords.htm