PROSODY IN URDU POETRY – A PHONOLOGICAL APPROACH

SARA HUSSAIN

ABSTRACT

In Urdu, written poetry and prosody bear the same relation as that present between poetry and poetic beauty. Prosody is the phonetic reflection of the textual verse. As in Urdu prose, poetry also inherits similar quantity sensitive nature, where moraic weights define verses. In poetry however, word boundary does not necessarily mark syllable division. Thus a verse is an amalgam of words with their phonetic content altered. Prosody dictates most of these alterations. A particular text of an Urdu verse can form various phonetic interpretations. This paper aims to uncover how this variant phonetic information is mapped to invariant phonetic tunes defined by prosody.

1. INTRODUCTION

Prosody or ilm-e-eroz is the study of rules which govern Urdu poetry (Badkhashani, 1969). These rules are able to form a reading pattern, which creates the essence of rhythm found in poetry. Rhythm or tunes so formed seems soothing to ears and contributes to the poetic beauty. However, they are not strictly part of the information, or message, which is conveyed by the poetry in general. Poets practice this art successfully without conscious knowledge of or ability to explain the principles they had followed (Riley, 2000).

This paper explores the phonological aspect of prosody in Urdu poetry. How prosodic rules are phonologically related to quantity sensitive Urdu language and what significance do moraic weight hold in this regard. The paper also tries to find how written verse in Urdu script is eventually mapped to prosodic rules of Urdu.

2. LITERATURE REVIEW AND PROBLEM STATEMENT

In (Hali, 2000, p.62) Hali asks the question whether prosodic weights are necessary for Urdu poetic verse. He argues that the essence of verse is independent of prosody. However, prosody in poetry enhances both its *beauty* and *effectiveness* (Hali, 2000).
This paper tries to find the roots of Urdu prosody in order to make some progress in discovering reason behind this poetic beauty.

Prosodic rules for Urdu poetry are explained in (Lakhnavi, 1921) and other such books found in literature. Each poetic verse is said in a rule called "*Behr"*. *Behrs* have different "wazan" or prosodic weights associated with them (Lakhnavi, 1921, p.52). In (Badkhashani, 1969) behr itself is defined as 'weight' of a verse. It refers to the meter set for poetry and is responsible for tunes found in poetry.

Weights in Urdu poetry take its root from Arabic. Khalil Bin Ahmed, an Arab, introduced ten Arabic words known as "erkan-e-ashira" (the ten elements) or "asol-e-afaeil". These pre-defined words are said to form weights and their sequence forms *Behrs*.

From these ten words sixteen Arabic and three Persian behrs are derived (Lakhnavi, 1921, p.55). The ten words can change to different predefined forms. New words so formed are called 'zehaf' or changed version of the word they originated from. Zehaf and their original words can coexist in a behr according to rules set and described in (Lakhnavi, 1921).

The 'ten elements' are themselves formed by a combination of three components called "seh gana". Each of the three components (namely 'sabab', 'watad', and 'fasala') is divided into two parts. These six types are formed by the combination of further two units: 'mutaharik' and 'sakin'. Description of 'seh gana' in terms of mutaharik and sakin is given in the appendix A. Erkan-e-ashira (ten elements) in terms of three components (sabab, watad, and fasala) are described in appendix B.

Mutaharik is defined as an alphabet with diacritic ‘zabr’ (short vowel sound of ə), ‘zeir’ (short vowel ɪ) or ‘peish’ (short vowel ʊ) associated with it. While an alphabet with diacritic ‘sakin’ [>] is called a sakin. Diacritic ‘tashdid’ [w] above a letter represents that letter as a sakin and then as a mutaharik. In Urdu script diacritics are not usually written but are assumed to be present and are always read.

Prosody of a single verse can be described by name of associated behr that gives detail of the number of elements (from erkan-e-ashira or zehaf) present and whether these elements come in their original form or have permissible changed words in it.

The other problem this paper intends to solve is to find some prediction in assigning predefine prosodic patterns (behrs) to different Urdu verses. Verse written in Urdu text if read in a particular commonly used pattern may be incorrect while the same ligature read differently may be correct.

3. METHODOLOGY

To understand prosody first step taken was to learn how it works. How behrs and erkan-e-ashira are mapped to the verses as they are read. So initial target was to practice assigning ‘weights’ given their true mapping. Numbers of verses practiced upon were 20 to 25.

As a second step, bottom up approach was used. Material found in literature was studied with detail analysis of (Lakhnavi, 1921) basic units of prosody as found in literature were listed. This information was mapped with knowledge of phonology.

Then, ten verses from the previously analyzed data were chosen and through trial and error an approximation, which they all follow, was drawn. All these verses were of type behr-e-kamil mesman salim, taken from (Iqbal, 1992, p.294).
To generalize and modify the algorithm so formed it was tested on various other types of verses. Selection of these verses was done to cover maximum types of behrs. One to three verses from seven different major behrs were analyzed (total major types of behr being 19). Almost all of these verses were selected from (Lakhnavi, 1921). But example given in this reference uses many words that are not commonly used today. So wherever another example was known it was used.

Lastly, verses that reflected features and behavior of vowels and diacritics were chosen. Most of these examples were taken from ‘Urdu lazmi’ a textbook for grade eleven and twelve. Also, verses containing words that undergo phonological deletion in Urdu were analyzed. Any available verse that contained such words was used for this purpose.

4. RESULTS

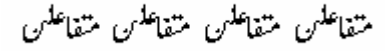
4.1 Basic unit of Urdu prosody

All of the erkan-e-ashira and their changed forms (‘zehaf’) could be broken down into fundamental unit of mutaharik and sakin. The table in appendix C shows distribution for the ten words of erkan-e-ashira.

On combining sequence of mutaharik + sakin i.e. forming sabab-e-khafif (for definition see appendix A), the following table in appendix D is generated. It is seen that by taking out sabab-e-khafif, only mutahariks are left.

4.2 Urdu vowels

Alphabets ‘alif’, ‘alif mad a’, ‘hamza’ and ‘noon ghonna’ of Urdu script are always associated with vocalic sound. Few other letters of Urdu at times act as vowels, while on other occasions they produce consonant sounds. These letters include alphabet ‘wao’, ‘yeh’, ‘ain’ and ‘gool hey’. These



characters can therefore be called as dual status letters. All the example verses presented in this section follow the following behr until otherwise given.

The following observation has been made about vowels of Urdu with reference to prosody.

Letter 'alif mad a': When this letter corresponds to a sakin it acts as a vowel 'alif' at the end of syllable. This syllable may even break word boundary as shown in the example below. Mapping of underlined portion to mutaharik (M) and sakin (S) is shown in table 1.

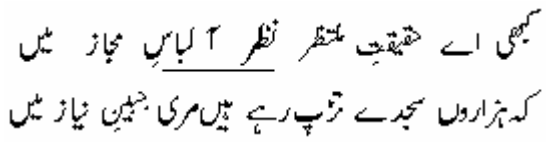


TABLE 1: Letter 'alif mad a' as sakin

ا	ب	ل	آ	ر	ظ	ن
ن	ل	ع	ا	ف	ت	م
S	M	M	S	M	M	M

On other occasions 'alif mad a' produces a long vowel sound of [a]. It is mapped to mutaharik as well a sakin (i.e. mutaharik + sakin). This is shown in example below. Table 2 gives detail mapping where cross sign shows deletion of the respective letter.

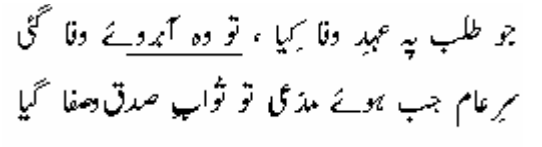
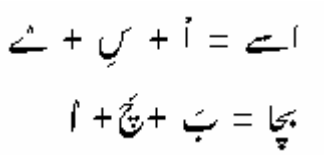


TABLE 2: Letter 'alif mad a' as long vowel

و	ر	ب		آ	ہ	و	و	ت
ن	ل	ع	ا	ف		ت		م
S	M	M	S	M	×	M	×	M

Letter 'alif': When alif comes as a mutaharik the diacritic associated with it (which may not always be present in text) determines its exact short vowel sound. However when 'alif' comes with a sakin, it indicates presence of diacritic 'zabr' (schwa) before it. Combination of zabr and 'alif' sakin makes a long vowel sound. Notice the following two words in which alif comes as a mutaharik and sakin.



Dual status letters: Letter 'ain' when occurring in middle or end of syllable it is pronounced as letter alif (a vowel).
Letter 'wao' and 'yeh' when associated with vowel sound maps with sakin. If however a mutaharik corresponds to the otherwise vowel wao or yeh, these letters are deleted i.e. they are not pronounced. Following verse shows deletion and occurrence of 'wao'.

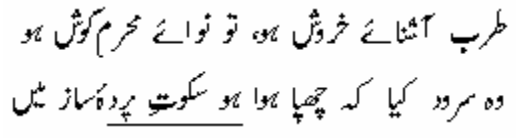


TABLE 3: Pronunciation of vowel 'wao'

ر	پ	ت	و	ک	س	و	ہ
ن	ل	ع	ا	ف	ت		م
S	M	M	S	M	M	×	M

Letter 'gool hay' is usually associated with consonant sound [h]. When this letter comes at the end of a morphological word it mostly represents a vowel sound in prose. In poetry at this position 'gool hey' can only be mapped to sakin. Collision of this 'gool hey' with mutaharik results in deletion of the letter. Consider the example and table below for the two conditions of presence and deletion of this letter.

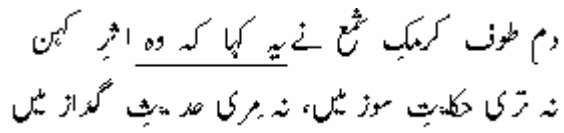


TABLE 4: Pronunciation of 'gool hey'

ہ	و	ہ	ک	ا	ہ	ک	ہ	ی
ن	ل		ع	ا	ف	ت		م
S	M	×	M	S	M	M	×	M

Letter 'noon ghonna': Presence of 'noon ghonna' in Urdu script indicates nasalization of the previous vowel. 'Noon ghonna' it self is not mapped to either mutaharik or sakin. Urdu prosody rules cannot determine presence or absence of this nasalization while reading a verse. However it is observed that nasalization never exists if the previous vowel is deleted. Behavior of 'noon ghonna' is given in example below.

وہ بتوں نے ڈالے ہیں وسوسے کہ دلوں سے خوفِ خدا گیا
وہ پڑی ہیں روزِ قیامتیں کہ خیالِ روزِ جزا گیا

TABLE 5: Nasalization produce by 'noon ghonna'

ا	ڈ	ے	ن	ں	و	ت	ب	ہ	و
ن	ل		ع		ا	ف	ت		م
S	M	×	M		S	M	M	×	M

TABLE 6: Deletion of 'noon ghonna'

و	ر	ں	ی	ہ	ی	ئ	پ	ہ	و
ن	ل			ع	ا	ف	ت		م
S	M	×	×	M	S	M	M	×	M

Letter 'hamza': In all the verses analyzed 'hamza' acts as a vowel sound that occurs after another vowel. The two vowels may or may not form diphthong. It is mapped to mutaharik as given in example below.

تو بچا بچا کے نہ رکھ اُسے، ترا آئینہ ہے وہ آئینہ
کہ شکستہ ہو تو عزیز تر ہے نگاہِ آئینہ ساز میں

TABLE 7: Letter 'hamza' as a mutaharik

ہ	ن	ی	ء		آ	ا	ر	ت
ن	ل		ع	ا	ف	ا	ت	م
S	M	×	M	S	M	S	M	M

4.3 Urdu diacritics

Almost all the diacritics present in Arabic script can be used in Urdu text. Description of those usually present in Urdu text is covered in this section.

Diacritic 'zeir': Diacritic zeir usually represents a short vowel in middle of morphological word. Notice the word [Ik] in the following verse.

بڑا شور سنتے تھے پہلو میں دل کا
جو دیکھا تو اِک قطرۂ خوں نہ نکلا

TABLE 8: Diacritic 'zeir' as short vowel

ط	ق	ک	اِ	و	ت
ن	ل	و ع			ف
S	M	S	M	×	M

Diacritic zeir is also used at the end of the word as 'alamat-e-ezafat' (indication of possessiveness). In which case it is either read as short vowel or long vowel [e] in poetry. A long vowel comes only when this zeir (alamat-e-ezafat) is matched with sakin. This is described by table below which uses example of table 4.

دمِ طوف کرمکِ شمع نے یہ کہا کہ وہ اثرِ کہن
نہ تری حکایتِ سوز میں، نہ مری حدیثِ گداز میں

TABLE 9: Diacritic zeir as long vowel

ا	ث	ر	ِ	ک	ہ	ن
م	ت	ف	ا	ع	ل	ن
S	M	M	S	M	M	M

Diacritic 'tashdid': By definition 'tashdid' indicates a sakin followed by a mutaharik sound of the letter it comes on top of. In poetry this diacritic is used in the same way as can be seen in the verse below (behr-e-ramal salim masbkh).

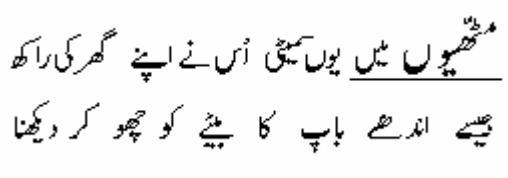
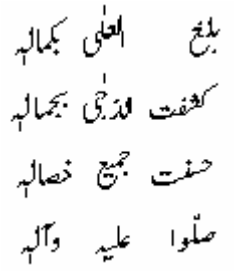


TABLE 10: Diacritic 'tashdid'

ں	ی	م	ں	و	ی	ٹھ	ٹھ	م
	ن	ت		ا	ل	ع	ا	ف
	S	M		S	M	M	S	M

Diacritic 'khari zabr' and 'Khari zeir': 'Khari zabr' produces a long vowel sound (similar to sakin alif) after the sound of alphabet it lies on. While diacritic 'Khari zeir' represents presence of an additional long vowel [i]. As seen from the following verses.



4.4 Behavior of consonant in Urdu words

On mapping Urdu text to prosodic words (erkan-e-ashira) few characteristics of Urdu words are noted.

Consonant clusters in onset position: Words with consonant clusters occurring in onset position of a syllable always maps to mutaharik. This can be seen in the word 'kya' in the verse below.

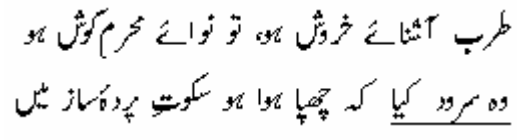


TABLE 11: Consonant cluster in onset position of a syllable

ا	ی	ک	د	و	ر	س	ہ	و
ن	ل		ع	ا	ف	ت		م
S	M		M	S	M	M	×	M

Consonant clusters in coda position: Such consonant clusters can be categorized to three cases. Notice the word 'lakh' in the given verse (behr-e-hizj masman ashtr).

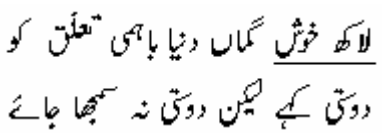


TABLE 12: Consonant after long vowel

ش	و	خ	کھ	ا	ل
ن		ل	ع	ا	ف
S	×	M	M	S	M

Mapping of two consonant cluster occurring in coda position of a syllable is shown by the word 'husn' in the following example (behr faelun mufaelun fa-ea).

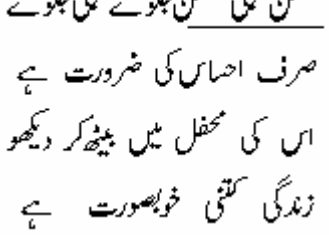


TABLE 13: Two coda consonants in a syllable

س	ح	ی	ہ	ن	س	ح
ن	ت	ا	ل	ع	ا	ف
S	M	S	M	M	S	M

Consonant cluster in coda position of a syllable can also be seen in the ‘nind’ by the following example of behr-e-rijz murabba. salim).

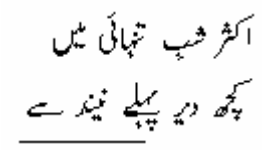


TABLE 14 Consonant cluster in coda position of a syllable

ے	س	د	ن	ي	ن	ے	ل
ن	ل	ع		ف	ت	س	م
S	M	M		S	M	S	M

Phonological deletion in Urdu: Certain Urdu words undergo phonemic deletion to produce phonetic sound associated with them. Deletion rules as found in Urdu prose are used in poetry too. Prosodic weights are assigned to the deleted text. Examples of few words that undergo deletion are as under:

خوش = خ + ش

خواہش = خ + ا + ہ + ش

خورشید = خ + ر + ش + ی + د

(may change to) اور = ا + ر

ہنس = ہ +nazalized ں

کھنچ = کھ +nazalized نچ

5. DISCUSSION

5.1 Moraic analysis of mutaharik and sabab-e-khafif

From result 1, it can be concluded that all the ‘erkan-e-ashira’ and their ‘zehaf’ can be written as a combination of sabab-e-khafif (mutaharik + sakin) and an independent mutaharik.

Syllabically mutaharik is a consonant (C) followed by a short vowel (V) or a single short vowel. Thus mutaharik always forms an open-ended syllable. Mutaharik can be seen as syllable structure CV and V that has one mora each. Thus from moraic analysis mutaharik can be defined as *an open-ended syllable with one mora*.

Syllabically sabab-e-khafif is usually taken as:

- Consonant (C) plus a short vowel (V) plus a consonant - CVC
- Consonant plus a long vowel (VV) - CVV
- Long vowel - VV

Thus from moraic analysis sabab-e-khafif can be defined as *a syllable with two moras*. Syllabic structure of Urdu word ‘kya’ is CCVV i.e. consonant cluster exists at onset position, thus it has two moras and fits well to the moraic definition of sabab-e-khafif so formed (as seen in table 11).

5.2 Problems

The major problem in mapping erkan-e-ashira to Urdu verses is the inability of the former to consider consonant clusters, which can be found in Urdu words. Reason for this inability is that all of the erkan-e-ashira and their zehafs do not possess consonant clusters. Moraic analysis such as presented above seems to solve this problem.

Few consonants that are mapped to mutaharik are phonetically pronounced as a sakin (table 12, 13 and 14). Observed distribution of these especial consonants is such that they always follow a sakin. One reason these consonants are mapped to mutaharik is the absence of sakin + sakin sequence in erkan-e-ashira and their corresponding zehafs.
From data analyzed it has been found that such consonant occurs only after a long vowel or coda consonant. Thus this particular consonant can very well lie in the coda position of previous syllable. But for some reason this does not happen in poetry.

Results show that any consonant C* occurring

1. After a long vowel VVC* is changed to CVV..C* (table 12)
2. After a short vowel and a coda consonant VCC* is changed to CVC..C* (table 13)
3. After long vowel and a coda consonant CVVCC* is changed to CVV..CC* (table 14)

Where '..' marks the boundary between sabab-e-khafif and mutaharik. This states that sabab-e-khafif is exactly equivalent to *two moras nothing more*. But why a coda consonant C* has been declared a mutaharik? One solution that can justify these mapping in above cases is to declare consonant C* as plus syllabic. With this conclusion consonant C* in (1), (2) and (3) will have weight one mora and can be called as mutaharik. This paper however does not experiment to confirm the presence of a plus syllabic feature in these consonants.

5.3 Predicting how to read an Urdu verse

Scanning textual verse from right, it maps to behr that defines it in two steps:

- Convert written text of poetic verse to reading pattern that eventually maps to behr. This will require breaking of written text into a list of individual characters. The list must also give an idea about how to read a character, i.e. when it is to be skipped, where it to be elongated or where it is to be shortened while reading.
- The second step is a simple one to one mapping between these characters of the list with prosodic structure of the verse.

The algorithm described in this paper tries to explore the correct way of reading a verse (i.e. find the 'list' as described above) given written text of Urdu verse and the behr it follows. It turns out that correct way to read a verse requires many deletions and even insertions to the otherwise prose text.

5.4 Limitations of prosodic rules

Not all the changes made in text to convert it to the proper reading pattern can be dedicated by prosodic rules. These changes include change in consonant spread (e.g. 'bh' may change to 'b'). Nasalization cannot be predicted by prosody either.

Phonological rules as found in Urdu language cannot be processed by prosodic rules. Rules (like that of assimilation and deletion) must first be applied on individual words of the verse. Then this sequence of pre-processed words can be mapped to prosody. The algorithm presented next inherits these limitations of prosody. In addition deletion of alif may occur. The algorithm is unable to check this change.

5.5 Input Requirements

1. Behr input is given in the form of actual words of erkan-e-ashira or their zehafs in the sequence they exist in that behr (e.g. mutafaelun mutafaelun mutafaelun mutafaelun). This format has been used in examples presented above.
2. Urdu text of the verse is broken into a series of individual characters before being fed to the system. Hamza is taken as a complete letter and is inserted before the ligature it lies above. Diacritics such as zeir are placed after the letter they occur with. Also the corresponding information that in this reference character exist as a consonant or a vowel. For all the cases seen this information can be extracted by pronunciation of the independent word. Information about consonant and vowel is especially vital for letters that have dual status.
3. The algorithm works only if the textual verse contains all the diacritics and letters whose occurrence makes a significant impact on the sound eventually to be produced. They include hamza, noon ghonna, diacritic zeir (occurring as alamat-e-ezafat), tashdid khari zabr and Khari zeir. Also, wherever the sound of noon ghonna occurs ligature noon should not be written, as found in (Lakhnavi, 1921).
4. Letter on which diacritic tashdid lies, on needs to be written twice.
5. If there is consonant cluster in the independent word occurring in onset position of a syllable it should be indicated.
6. For words like 'khahish' (wish) and 'khosh' (table 12) in which deletion of

wao occurs at word level, the deleted text should be fed to the system.

5.6 Algorithm

Step 1: Each word from input behr is broken to its corresponding combination of mutaharik and sakin (from Result 1). A count of total elements (mutaharik and sakin) is stored in a variable say A. On each call for next behr of element, next element (mutaharik or sakin) in the sequence starting from the first such element, is returned.

Step 2: A count of total letters (hamza and noon ghonna included) is stored in a variable say B.

Step 3: If B-A is positive
Then required deletion = B-A
required insertion = 0
Else
required insertion = A-B
required deletion = 0

Step 4: Set deletions made = 0
Insertions made = 0
Loop count = A

v = element from verse
w = element from behr (it can either be mutaharik M or sakin S)
('+' and '- ' sign below is used to get the element present in number calculated)

The following conditions are repeated till count is not equal to zero. Count decrements after every iteration.

Condition 1:
Consonant clusters should be inserted in LIST with no mark of M or S. The last consonant in this cluster should be marked as M. Consuming one element from behr and all of the consonant cluster from elements of verse list.
Condition 2:
If v = alif mad a
If w = M and w+1 = S
Output v as long vowel [a] mapped with M and S

Insertions made = insertions made +1
w = w+2 consume two elements of behr
v = v+1 consume one element of verse
Else (w = S)
Output v as short vowel mapped to S

w=w+1 (one to one mapping)
v=v+1
End If
End If

Condition 3:
If v = 'wao' as vowel
If w = M
(Long vowel sound 'wao' does not exist)

Output to LIST at v-1 as having peish

deletion made = deletion made +1
v = v+1 consume one element of verse

Else (w = S)
Output to LIST v as long vowel mapped to S

w=w+1
v=v+1
End If
End If

Condition 4:
If v = 'yeh' as vowel
Similar to condition 3 except if yeh is deleted a zeir is placed at v-1

Condition 5:
If v = 'gool hay' as vowel
If w = M
(Vowel sound 'gool hay' is deleted)
deletion made = deletion made +1
v = v+1 consume one element of verse

Else (w = S)
Output to LIST v as short vowel sound of alif mapped with S
w=w+1
v=v+1
End If
End If

Condition 6:
If v = 'hamza' and w = M
Output to LIST v as short vowel sound of alif mapped to S
w=w+1
v=v+1
Else
Error has occurred. It may be that 'hamza' is placed in text when its sound is not present.

v = v+1 (ignoring hamza)
End If

Condition 7:
If v = 'noon ghonna'
If v-1 is not a long vowel
deletion made = deletion made +1
v = v+1 consume only 1 element of verse
End If
Else
It is an ambiguous state either It can produce nasalization in previous vowel or can be deleted. In both cases
v = v+1 consume 1 element of verse only

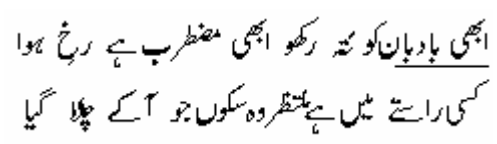
Condition 8:
If v = 'ain' as a vowel
It is processed as alif.

5.7 Conditions for diacritics

Condition 9:
If v = 'zeir' (as alamat-e-ezafat)
If w = S
Output v as vowel sound mapped to S
insertion made = Insertion made +1
w = w+1 consume 1element of behr

Else
Output to LIST at v-1 as having 'zeir'
v=v+1 ('zeir' is consumed)
End If
End If

Condition 10:
If v = 'khari zabr'
If w = S
Output v as vowel sound of alif mapped
To S
(Insert long vowel of alif)
w = w+1 (consume one element of behr)
v = v +1 (consume 'khari zabr')
Else
Error. Cannot process 'khari zabr'.
v = v+1 (ignoring it)
End If
End If



Condition 11:
Diacritic 'khari zeir' is processed similar to 'khari zabr' while producing a long vowel sound of [i] (letter 'small yeh').

Condition 12:
All other diacritics are ignored i.e. v = v +1

Condition 13:
All other letters of verse are matched one to one with the elements of the behr.

Step 5: It should be noted that every time deletion made goes beyond the required number req. insertion is incremented and similar is the case for insertion made.
If deletions made = req. deletions, insertions made = req. insertions and both input from behr and verse are exhausted Then LIST gives the format to read the verse. Else the behr does not map to the given verse.

5.8 Implementation and Testing

Most difficult part to implement is the input engine that takes written verse and converts it into a list of individual characters (with vowels indicated) and diacritics. This requires automation of phonological rules with respect to Urdu dialect. In prosody there seems to be an agreed dialect to be used.

An easier implementation is to use a human user to input the text format after applying phonological rules. Input engine may then indicate consonant clusters and process diacritic 'tashdid'. Diacritic zeir when occurring at the end of morphological word should be marked as 'alamat-e-ezafat'.

List of behr (in form of mutaharik and sakin) can be stored in a LIFO (stack) data structure. List of processed letters and diacritic with their corresponding information can be stored in a similar structure. Output can be given in a list structure that gives the letter (or diacritic) and it's mapping with mutaharik or sakin. If the correct reading pattern is known, then on matching output list to this pattern can test the algorithm.

5.9 Example

A dry run of the algorithm is shown below. Consider the underlined portion of the verse

below that follows the behr mtafaelun mtafaelun mutafaelun mutafaelun.

Total number of elements of behr A =28
Total number of elements of verse B =31
Number of deletions required = 3
Loop count = 28

First two letters are processed by condition 13. By condition 4 letter ‘small yeh’ is deleted and a diacritic zeir is placed below previous letter ‘bh’. Number of deletion made is incremented. The next four letters are simply mapped by condition 13. The output list shows like this :

Table 15: Output list of the algorithm

ا	ب	د	ا	ب	ھ	ا
S	M	M	S	M	M	M

6. REFERENCES

1994.Urdu Lazmi kyarvin barvin jamaton ke liye. Maktaba Moin-ul-adab, Lahore

Badakhshani, Mirza Maqbol Baig, 1969. Urdu lokhait. Markazi Urdu Board, Lahore

Hali, Altaf Hussain, 2000. Moqadma sher-o-shaery ma tanqidi jaeza. Popular Publishing House, Lahore

Iqbal, Mi. 1992. Bang-e-dara. National Book Foundation, Islamabad

Lakhnavi, Yas Azim Abadi, 1921. Charakh-e-sokhan. Monshi No Lakshor, Lakhnao

Riley, Brain T. 2000. Problems and methods in linguistics. Cosmo Publications, New Delhi

APPENDICES

Appendix A

Definition of 'seh-gana' in terms of mutaharik and sakin.
M stands for mutaharik
S stand for sakin

Table A.1: 'seh-gana' in terms of mutaharik and sakin

A	M	S			
B	M	M			
C	M	M	S		
D	M	S	M		
E	M	M	M	S	
F	M	M	M	M	S

Alphabets A, B, C, D, E and F above corresponds to 'seh gana' and their types their names are:

A.سبب خفیف B.سبب ثقیل C.وتد مجموع D.وتد مفروق
E.فاصلہ صغریٰ F.فاصلہ کبریٰ

7.2 Appendix B

Definition of erkan-e-ashira in terms of 'seh-gana'
Letters in the table below represents the following:

A.سبب خفیف B.سبب ثقیل C.وتد مجموع D.وتد مفروق
E.فاصلہ صغریٰ F.فاصلہ کبریٰ

Table B: 'erkan-e-ashira' in terms of mutaharik and sakin

1	C	A	
2	A	C	
3	C	A	A
4	A	C	A
5	A	A	C
6	D	A	A
7	A	D	A
8	A	A	D
9	E	C	
10	C	E	

The numbers above stand for the following erkan-e-ashira

1.فعولن 2.فاعلی 3.مفاعیلی 4.فاعلاتن 5.مستفعلن
6.فاع لاتن 7.مس تفع لن 8.مفعولاتن 9.متفاعلی 10.مفاعلتن

7.3 Appendix C

Distribution of 'erkan-e-ashira' in terms of mutaharik and sakin
M stands for mutaharik
S stand for sakin

TABLE C.1: 'erkan-e-ashira' in terms of mutaharik and sakin

1	M	M	S	M	S		
2	M	S	M	M	S		
3	M	M	S	M	S	M	S
4	M	S	M	M	S	M	S
5	M	S	M	S	M	M	S
6	M	S	M	M	S	M	S
7	M	S	M	S	M	M	S
8	M	S	M	S	M	S	M
9	M	M	M	S	M	M	S
10	M	M	S	M	M	M	S

The numbers in the previous table stand for the following erkan-e-ashira

1.فعولن 2.فاعلی 3.مفاعیلی 4.فاعلاتن 5.مستفعلن
6.فاع لاتن 7.مس تفع لن 8.مفعولاتن 9.متفاعلی 10.مفاعلتن

7.4 Appendix D

Distribution of 'erkan-e-ashira' in terms of mutaharik and sabab-e-khafif
M stands for mutaharik
K stands for sabab-e-khafif

Table D.1: 'erkan-e-ashira' in terms of mutaharik and sabab-e-khafif

1	M	K	K		
2	K	M	K		
3	M	K	K	K	
4	K	M	K	K	
5	K	K	M	K	
6	K	M	K	K	
7	K	K	M	K	
8	K	K	K	M	
9	M	M	K	M	K
10	M	K	M	M	K

The numbers in the previous table stand for erkan-e-ashira.