

Urdu Encoding and Collation Sequence for Localization

Sarmad Hussain¹, Sana Gul, Afifah Waseem
Center for Research in Urdu Language Processing
National University of Computer and Emerging Sciences

1. Introduction

Information has become such an integral part of our global society that its access is considered as a basic human right [1, 2]. Access to this information may rightly be equated with access to the Information and Communication Technology (ICT) specifically the Internet that is believed to be the dominant carrier of information across the globe. Currently, English is the lingua franca for ICTs with the majority of content being in English. This makes English language information available on ICTs inaccessible to the majority of world population that cannot read or write in English. This is especially applicable to Pakistan where only 5-10% of the total population can understand English [3, 4]. Due to this low English literacy rate the ratio of internet usage in the country has been adversely affected. According to one estimate, at present only 1.2 % of the total country population is using internet [4]. Evidently, unless the ICTs are available in languages that are understood by the masses, the majority of the country's population would remain deprived of the information available through the English-centric ICT devices.

Enabling ICTs in the local language of the user is known as "localization". Specifically, it is enabling computing experience in linguistic culture of the user. Linguistic culture is not just limited to the language but how the language is used by the environment of the user. Thus, for Punjabi speakers in India, the computer should display the language in Gurmukhi script and for Punjabi speakers in Pakistan, the same language should be displayed in Arabic script.

Localization of ICTs requires local language applications based on local language computing standards. These standards include character set encoding, keyboard (and keypad) layout, collation/sorting sequence, locale and ICT terminology (overview of these standards is given in [5, 6]).

In addition to definition of standards, applications also need to be developed for local language computing to support script, speech and language processing. Within these areas, there is a large variety of applications required, some being more fundamental in nature, while others are more advanced and complex but equally vital for end users. These applications include definition of input methods, normalization and sorting utility [e.g., 7, 8], find/replace utility, fonts, lexicon, thesaurus, spell checker, grammar checker, text-to-speech system, speech recognition system, machine translation system and optical character recognition system [5, 6]

Currently work is being done to develop these standards and applications for Urdu language, the lingua franca of Pakistan. This paper focuses on the evolution and standardization of Urdu language character set encoding. In addition, this paper also describes the issues and development related to Urdu collation sequence and proposes a set of collation elements to enable it. Both standards are essential for the development of Urdu computing.

2. Urdu Encoding

Character encoding may be defined as assigning a unique number to each language character to be processed by the computer. Whenever a character is input from keyboard or other input devices, this particular code is generated internally in the computer. Arbitrary encoding may be defined for any application (e.g. 80 for letter 'a', 81 for letter 'b'). However, if different vendors are

¹ Associate Professor, National University of Computer and Emerging Sciences, B Block, Faisal Town, Lahore, Pakistan. Email: sarmad.hussain@nu.edu.pk. URL: www.crup.org.

defining arbitrary encodings, their encodings may not agree with one another. This would mean that the data files generated by their application(s) will be accessible only through their applications and may not correctly represent data in another vendor's software (e.g. another vendor may define 80 to be letter "P" and 81 to be "Q"). Though initially application developers were able to get away with this, even successfully use this mechanism to tie in users, with the advent of the Internet, it has now become increasingly essential to standardize the encoding scheme because users are accessing data created by a variety of sources through web browsers (a single application). Realizing the significance of standardizing encoding, work was done early for English and American Standard Code for Information Interchange (ASCII) was defined in 1968. This standard had 128 slots defined using 7 bits by American National Standards Institute (ANSI). However, this code chart only contained English characters. This table was extended to 8 bits to include other Latin based languages spoken in Europe and South America. Later international organizations like International Standards Organization (ISO) also adopted ASCII as ISO/IEC 646 standard. 8-bit code pages for other languages were also introduced as international standards ISO/IEC 8859 -x (e.g. ISO/IEC 8859 - 6 for Arabic).

Many different encoding schemes have been developed for Urdu as well by various vendors. However, no national or international effort was undertaken to standardize Urdu encoding, resulting in difficulties in exchanging Urdu data. The necessity of a standardized character set was first highlighted in Oct, 1997 at the 4th National Computer Conference, Islamabad, organized by the Pakistan Computer Bureau [9]. To formulate these standards a one-day long seminar on "The Standardization of Urdu Keyboard Layout and Internal Character Representation" was conducted at Foundation for Advancement of Science and Technology (FAST) Institute of Computer Science, Lahore, in 1998. This seminar resulted in the formation of four separate committees focusing on Urdu Code Page, Urdu Keyboard Layout, Urdu Email and Urdu Internet. As the work on the last three committees was based upon the work accomplished by Urdu Code page team, therefore the development of Urdu Code page was prioritize over others. The recommendation through this committee resulted in the formulation of the *Urdu Zabta Takhti* (UZT 1.01; see [10] for details) which was accepted as 8-bit encoding standard for Urdu by Government of Pakistan in 2000 [11]. This standard is briefly described below (for a detailed discussion see [12]).

2.1. UZT 1.01 [12]

UZT 1.01 is an 8 bit code page and can accommodate 256 slots for character representation. This standard has all the characters identified to do publishing for Urdu. Though it is primarily an encoding standard (and sorting is usually not catered directly in such standards, e.g. see Collation Section below), the standard was devised to do some implicit sorting based directly on encoding. It has been divided into various logical sections that are as follows:

- i. The Control characters
- ii. Punctuation and arithmetic symbols
- iii. Digits
- iv. Urdu aerabs/diacritics
- v. Urdu characters
- vi. Reserved control space
- vii. Special symbols
- viii. Reserved expansion space
- ix. Vendor area
- x. Toggle character

UZT 1.01 is shown in Figure 1 below. A brief description of these logical sections is also given.

High Hex Digit

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0			Sp	۰	@	ا	ژ	م			لله	○	[
1			!	۱	H S	آ	ز	ن			جا		\			
2			"	۲	ہمزہ اضافہ	آ	ژ	ن			بسم اللہ]			
3			#	۳	گروہ اضافہ	ب	س	و			۴		U s			
4			C	۴	۱	پ	ش	ژ			۶		{			
5			۵	۵	۱	ن	ص	ہ			۶					
6			&	۶	۴	ن	ض	ة			۶		}			
7			,	<	۱۹	ن	ط	ء			ع		D a			
8			(۸	۱۱	ج	ظ	ی			۱					
9)	۹	۱۱	ج	ع	ن			۶					
a			*	:	۱۱	ح	غ	ہ			۶					
b			+		۱۱	خ	ف	×			۶					
c			,	<	ط	د	ق	۱			۶					
d			.	=	۰	ڈ	ک	۱			نخ					
e			D c	>	۷	ذ	ک	۱			۱۱					→
f			D c	؟	۱۵	ر	ل				۵					

Figure 1: Urdu Zabta Takhti 1.01

Control Characters

Slots from decimal positions 0 – 31 and 127 are dedicated to the control characters which include characters like null (0), new line (10), escape (27), delete (127), etc. As these characters are used universally, UZT has used the same slots for these characters as the ASCII characters, in order to avoid conflicts.

Punctuation and Arithmetic Symbols

These symbols have two dedicated places from slots 32 – 47 and 58 – 65 assigned for punctuation and arithmetic symbols respectively. This ordering is similar to the ASCII. Some symbols are logically similar however transformed into a different shape. Finally a ligature break (65) has been included and distinguished from word break (32). The Urdu script is although connected, however breaks between characters may occur within words or across words as illustrated in Figure 2. To distinguish word boundary from ligature boundary hard space and space are used respectively where through former a single ligature can be broken into two ligatures without breaking the word, while the latter breaks ligatures into different words.

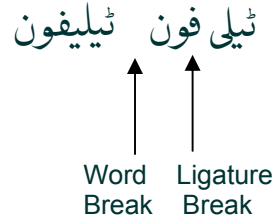


Figure 2. Word break and Ligature break in Urdu

Digits

Urdu numerals from zero through nine are included in slots 48 – 57 same as ASCII.

Urdu Diacritics (Aerab)

The aerab are included from slots 66 – 79 and slots 123 – 126. These aerab cover basic Urdu vowels Zer, Zabar and Pesh and also extended aerab set (e.g. Khari Zabar, Tashdeed, etc.). Infrequent aerab like Noon Ghunna sign were also included to cater to the publishing needs of Urdu.

Urdu Characters

Urdu characters fill slots 80 – 121, therefore there are 42 characters included. 'Do-chashmey hay' is included at the end of the list in slot 122 (instead of its traditional position adjacent to 'Gol hay' in slot 117) for two reasons. First, though written separately, 'Do-chashmey hay' is not a character of Urdu but forms characters when combined with other characters (e.g. /b^h/, /p^h/, etc.) and therefore needs to be distinguished from characters. Second, the sorting sequence, which requires all words starting with /b/ to be before all words starting with /b^h/ (and similarly with other unaspirated-aspirated consonantal pairs), comes out naturally if 'Do-chashmey hay' is placed at the end of the character list.

Reserved Control Space

ASCII is a seven-bit standard, from 0-127. Earlier systems, of which are possibly still deployed, still conform to this standard. Therefore, if one byte code is sent to these systems, they truncate the most significant bit and convert the byte code to seven-bit code. This would be especially dangerous if slots 128 – 159 are used, because truncating the most significant bit would map these slots onto the control characters resulting in unpredictable behavior. Therefore, these slots have not been used. For the same reasons of truncation, all the Urdu characters, aerab and digits have been included in lower 128 slots. This ensures that even if a system conforms to a seven-bit code, it may still effectively use UZT to store information in Urdu.

Special Symbols

Slots 160 – 176 include these symbols, with room for further expansion after it. Slots 192 – 199 include more general symbols also in ASCII, including square brackets, curly brackets, underscore and dash.

Reserved Expansion Space

Slots 177 – 191, 200 – 207 and 240 – 253 have been reserved for future expansion. The committee devising Urdu standards may fill these slots in the future. The slots have been left vacant in anticipation of the future needs of Urdu.

Vendor Area

The code page covers the general and widely used Urdu language characters, aerab and symbols. However, there may be special needs that may arise for formatting or for including other more specialized characters not commonly used in Urdu. Slots 208 – 239 have been reserved for this purpose.

Toggle Character

Finally, a character in slot 254 has been reserved to toggle between various code pages. Currently toggle character followed by character zero (slot 48) would mean start of UZT 1.01 and toggle character followed by one (slot 49) would mean ASCII. This will be relevant for the <filename.utx> (Urdu Text File) format.

2.2. Unicode

Initially most documentation was done in a single language, therefore 8-bit single language code pages served the need. However, in 1990s, with increasing needs for multi-lingual documents (where one could require Japanese and Arabic in the same document), it was realized that defining 8-bit code pages were not a scalable solution. Adding code pages for various languages and scripts and using them together in one application created a lot of difficulty and complexity in processing because users had to keep toggling between them.

To address this issue, major vendors got together and created Unicode consortium (www.unicode.org). This consortium started working on developing a singular, unified and universal code chart which would contain all characters of all languages. As 8-bit (256 slots) code pages were insufficient for this requirement, Unicode character encoding standard was developed using 16 bits (65536 slots) [13]. This space has been divided to cater to various scripts and thus bypassed the need for toggling for different languages. Unicode provides a consistent way of encoding plain text in multiple languages within a single document and brings in a simple, efficient manner. Unicode standard has also been adopted by ISO as ISO/IEC 10646, and is now jointly monitored and updated by Unicode consortium and ISO/IEC JTC1/SC2/WG2, the working group assigned to look at character encodings.

As Unicode is a script based standard, Urdu support in Unicode is given in Arabic Script block from Hexadecimal 0600 till 06FF (because Urdu is written in Arabic script along with Farsi, Pashto, Sindhi, etc.). In addition, there are two more blocks for Arabic, which were added to provide backward compatibility and have now been deprecated. These blocks are from Hexadecimal FB50-FDFF and FE70-FEFF (except FDFx sequence, which is still in use and contains ligatures). Only basic level work was done for Urdu when UZT 1.01 was formalized. UZT gave a good measure of Urdu requirements. Using this as a baseline, gap analysis was done between UZT and Unicode [14]. The gap analysis resulted in proposal for adding missing Urdu characters to ISO/IEC 10646 standard, which was submitted to ISO/IEC JTC1/SC2/WG2 meeting in Dublin in 2002 for consideration [15]. After the addition of characters through this proposal and other proposals presented by other ISO/IEC JTC1/SC2/WG2 members, Unicode 4.0 has complete support of Urdu language.

As Unicode standard has to cater to multiple existing systems and multiple languages within a script, redundancies are introduced in it. However, sometimes these redundancies are visual and not present functionally (this is not always evident from the glyphs given in the code charts, but is explained where character names, descriptions and behaviors are elaborated in [13]; e.g. for Do-Chashmey Hay, there are two codes, Hexadecimal 0647 and 06BE, but latter has been recommended for use in Urdu; also see 0649 and 06CC). Therefore, before using Unicode, it is necessary to determine (and perhaps standardize nationally) which codes in Unicode need to be short-listed for Urdu. In addition, some normalization also needs to be done to address the redundancies introduced through the standard (e.g. Hamza Alif = Alif + Hamza above: Hexadecimal 0623 = 0627 + 0654; also 0622 = 0627 + 0653) [7].

Figure 3 below is a subset of the Unicode code chart for Arabic (available from <http://www.unicode.org/charts/PDF/U0600.pdf>) which identifies the minimal set of Urdu relevant characters. These characters include most of the characters in UZT 1.01 discussed in the previous section. The characters which have been “blacked-out” from the original chart either belong to other languages (e.g. Sindhi, Pashto) and not used in Urdu or include characters which can be derived from the characters shown (e.g. Hamza Alif = Alif + Hamza above).

	060	061	062	063	064	065	066	067	068	069	06A	06B	06C	06D	06E	06F
0				ذ												۰
1			ء	ر	ف					ڑ			ہ			۱
2			آ	ز	ق									ک		۲
3				س									ة			۳
4				ش	ل									-		۴
5				ص	م											۵
6				ض	ن					چ						۶
7			ا	ط												۷
8			ب	ظ	و					ڈ	ژ					۸
9				ع				ٹ			ک					۹
A			ت	غ			٪					ں				
B			ث					ر								
C			ج					د					ی			
D			ح													
E	م		خ					پ				ھ				
F		؟	د								گ					

Figure 3: Urdu Character Subset for Unicode Arabic

In addition to these another important character used for Urdu is the Zero Width Non-Joiner (ZWNJ) Unicode 200C (functionally equivalent to Ligature Break in UZT 1.01; see Figure 2). Also, there are ten ligatures from the Unicode FDF2-FDFB that are included.

Unicode standard provides almost complete support for Urdu. However, there are still few deficiencies. Hamza is one such example. In Unicode, Hamza character ؤ is declared a non-joiner (i.e. it does not connect with letter following it). However, in Urdu language words like قائل require Hamza to join with characters following it. Similarly Bari Yay ے is also considered a non-joiner in Urdu (with the following character), but word ے کار is also commonly written in Urdu as ےکار. These issues still need to be resolved with Unicode standard to complete Urdu support.

3. Urdu Collation

After encoding, definition of collation sequence for a language has perhaps the most important requirement for standardization. Any significant data processing (e.g. maintaining employee records, voter lists, etc.) requires lexicographic sorting of textual data, which is usually different from sorting on the codes assigned to characters of the language. For example, if English is sorted on ASCII code, it would sort an alphabet string to "ABCDE...Zabcde...z". However, lexicographic sorting requirement of English is "AaBbCcDdEd...Zz".

In addition, unlike Unicode encoding, which is a script based standard, collation is usually specific to a particular language. This would mean that different languages, which are based on the same script may still have different sorting sequences. These differences arise because languages base sorting on many different factors including characters, capitalization, marks/diacritics and strokes [17]. For example, Urdu and Sindhi are both based on Arabic script, but have different

character level sorting sequences: ب پ ث ش (for Urdu) and پ ث ش ب (for Sindhi) [16].

Sorting order differences can also be demonstrated through examples from the Latin script languages. For example sorting sequence of the same words is shown for English and French diacritic level differences in sorting conventions in Figure 4 (for a detailed discussion see [17]).

English	French
casa	casa
chaque	chaque
choisi	choisi
choy	choy
císta	císta
Ciurika	Ciurika
cote	cote
coté	côte
côte	coté
côté	côté

Figure 4: English and French Sorting Conventions [17]

As Unicode encoding standard is script based, where only one sequence of characters may be stored for all languages using that script, it is not possible to realize varying sorting sequences within it. Therefore, separate codes (based on languages) different from Unicodes are used to do language-centric sorting. These codes are referred to as collation elements. In addition, to cater to the various levels of sorting (depending on character, mark, capitalization, etc.) these collation elements are divided into multiple levels (see [8] for details). A default collation element set has also been standardized (not for any particular language) and may be used as a starting reference [32].

This section first discusses the issues and their resolution to develop Urdu lexicographic standard for Pakistan. The section then defines collation elements of Urdu compatible with Unicode collation algorithm as defined in [7, 8]. This section also presents results of implementation of the algorithm using the proposed collation elements to show that they can be successfully used to sort in Urdu.

3.1. Issues with Urdu Collation [18]

Initial work was done to identify Urdu collation sequence by studying multiple dictionaries available for Urdu. Results showed that multiple conventions were being used and no one sequence was agreed as final [18]. The issues discovered are summarized below.

Issue # 1: Ambiguity between the sequential position of آ and ا

Data gathered from eight different dictionaries of Urdu language, depict the following inconsistency in the sequential position of the Urdu characters آ and ا. Data from [20], [24],

[25], [26] shows that آ follows the basic ا in a sorted sequence. The sorting sequence would be:

ا آ آ ا (from right to left). Justification for this order is that historically آ was considered

to be a different stylist way of righting double ا i.e. two ا occurring in a row [27]. Therefore it was

postulated that linguistically آ does not exist as a separate character. Another sorting sequence

for ا and آ can be seen from the data gathered from [19]. This dictionary recognizes آ as a

separate independent character of the language, and places it sequentially before ا. The sorting

order thus formed is: ا آ آ ا (from right to left)

This sequence is a more widely used sorting order convention in Urdu language. Yet another sorting order for the same characters can be seen, by gathering data from [21], [22], [23]. Here

as in [19], linguists recognize آ as a separate character, contrary to [20], [24], [25], [26], but order

sequentially after | contrary to the rules in [19]. The sorting order thus formed is as follows:

ا ا ب آ آ ب (from right to left).

Issue # 2: The ambiguity of digraphs formed with Do-Chashmey Hey (ھ)

The ambiguity in determining the sequential position of ھ when it is combined with other basic Urdu characters (e.g. ھب = ب + ھ). The question is whether this digraph is a separate character or a sequence of ب and ھ where ھ comes before/after ب. This discrepancy in the sorting order of Urdu language can be observed while examining data in Urdu dictionaries. A set of dictionaries including [21], [22], [23] believe that the composite characters like ھب formed after the joining of a ب + ھ, is an independent character. They treat ھب as a separate character and position it after the occurrence of ب sequentially, giving the following sorting sequence:

بھنگلی بھنگلی بیٹی بھابھی بھنگلی (from right to left). Another Urdu dictionary [20] treats the digraph ھب, not as an independent entity but a sequential occurrence of the characters ب and ھ. Here ھ in contrary is considered as a separate and independent character. This dictionary sorts the two characters, ب and ھ at the same level in the first pass, but gives precedence to ھ over ب in the second pass of sorting. The sorting sequence thus formed is: بھابھی بھنگلی بھنگلی بیٹی (from right to left). Similar treatment of the digraph ھب, is observed in [19], [24], [25], [26] as it was observed in [20], where the digraph is treated as a sequence of ب + ھ, however, these dictionaries give precedence to ب over ھ in the second pass of sorting. The sorting sequence so formed is: بھابھی بھنگلی بھنگلی بیٹی (from right to left).

Issue # 3: Ambiguity in the sequential position of ں and ِ

There are two major issues regarding the morphological and phonetic characteristics of the Urdu ِ. Firstly, status of ِ as a separate, independent character is very ambiguous. This ambiguity arises because of its phonetic behavior. Observations on the use of ِ in Urdu language shows that this character adds “nasalization” feature to the preceding vowel with which it occurs in a word. E.g. گاؤں، کماں at the same time it does not produce a separate sound. However, ں is a nasal constant itself. For this reason few dictionaries consider ِ as a variation of the Urdu character ں. The second major issue is pertaining to the sorting order of the characters ں and ِ. One set of dictionaries [19], [21], [20], [26], [25], and [24] gives precedence to ِ in the sorting order following the sequence (from right to left) ماں مان, while another set of dictionaries [22], [23] give precedence to the Urdu character noon over the Urdu character noon gunnah in a sequentially sorted sequence for example مان ماں.

Issue # 4: Ambiguities pertaining to the sequential position of ے in the Urdu language sort order

The Urdu character ے is borrowed from Arabic language. In Arabic Tay sound is represented by ے when it is part of the root morpheme. However, if the sound occurs as part of feminine or other suffix, Tay sound is represented by ے. In Arabic, it is pronounced as ے whenever there is a pause observed after the word ending with ے. However, this phonetic property of the Gol Tay does not exist in Urdu language. In [28] following sequence of the words is observed: دائرہ then دائرۃ المعروف. This sequence shows that here Gol Tay is placed sequentially after the Gol hay. However, in [29] the word دائرہ follows دائرۃ المعروف.

Issue # 5: Ambiguities related with status of ء

In certain cases Hamza takes a character joiner but in other cases, it resides directly on another

character. For example, in words like قائل Hamza takes an explicit glyph joiner (compare with

جراث، پلاؤ where Hamza is not written), but does not take a joiner for the following words:

جراث، پلاؤ (compare with جراث، پلاؤ where Hamza takes an explicit joiner), Characters always have a joiner and aerab never have a joiner. Thus, Hamza shows a mixed behaviour and as both types have differing sorting behaviour, it is also unclear how to sort Hamza.

Issue # 6: Ambiguity between the sequential position of ی and ے

Though it is fairly conventional and agreed upon by all dictionaries that ی comes before ے, nevertheless few dictionaries do not agree whether this difference in collation is in the first or second pass of sorting, giving the following variations. Sorting sequence of three words is given in two different ways by two sets of Urdu dictionaries. According to [19], [21], [22], [23], [26] the sorting sequence from right to left is: بی بی بی ے while the same words are sorted as

بی بی ے according to [20], [24], [25]. In addition there exists another ambiguity

regarding the sorting sequence of yeh when it is written within a word, e.g. in the word بیابان it is unclear whether to sort the middle Yay as a ی or a ے.

Issue # 7: Sequence for aerab: Zer, Zabar, Pesh, khari Zabar, tashdeed etc.

The words with diacritic marks or aerab are processed at the second level of the sorting process. However it has to be determined how they are mutually sorted and that whether words without them are to be placed sequentially before or after them, e.g. sorting order of words like بن، بن، بن has to be determined.

3.2. Recommendations for the Resolution of Issues with Urdu Collation

Work had been in progress to finalize a single collation sequence, which could be standardized for Government of Pakistan. The above mentioned problems were first highlighted in 2002 [30]. Later in 2003 these issues were reviewed in a meeting held at Center for Research in Urdu Language Processing at National University of Computer and Emerging Sciences. Finally, the issues were resolved in the final meeting held at National Language Authority on 26th January, 2004. The following are the recommended resolutions of the issues discussed.

Before individual issues were taken up, for consistency some basic rules of thumb were agreed for standardizing the collation sequence. They were then used to form the recommendations in a consistent manner. These rules were:

1. Given two characters, one being core and other being secondary in nature and not conventionally considered as core, the former would precede the latter in sorting order.

2. As core Urdu characters are grouped according to their shapes (and not sounds), any non-primary characters should be introduced according to the shape and not their sound.

Following these two basic principles the ambiguities pertaining to the conflicting sorting orders were resolved as explained below.

Resolution of Issue # 1

آ is taken as an independent character and is no longer considered as a variation of ا.

Justification behind declaring آ as a separate character is because it is very productive in Urdu and now native speakers' intuition treats it as different from a sequence of two ا (most likely now this stylistic variation has been lexicalized). آ is placed sequentially after ا in the sorting order using the two rules of thumb discussed above.

Resolution of Issue # 2

اُ is not a separate character of Urdu, but forms a digraph indicating the aspirated version of the associated consonant. As these digraphs form different phonemes and are very productive in their usage, following earlier work [23] they were taken as separate characters and placed sequentially after their basic character, e.g. in the اُت will follow اُت.

Resolution of Issue # 3

Noon gunnah was agreed to be separate from noon, and using the two rules, placed after noon for sorting.

Resolution Issue # 4

Gol Tay was also identified as a separate character. Although it produces a Tay sound but following the two rules, it was recommended to be placed after ا.

Resolution Issue # 5

Hamez ه takes dual form, as a combining mark and a character. It was recommended that with vowel characters ا، و، ی، ے، it should be considered as part of aerab, but outside this context it is taken as a character. When it appears as a character, the conventional sorting sequence of ه is

maintained, where it appears after ۞ but before ی. When it occurs as an aerab, it follows the sorting sequence discussed in Resolution of Issue # 7 below.

Resolution of Issue # 6

Both ی and ے are taken as characters and thus are to be sorted at the first level. Additionally, it was clarified that diction rules of Urdu state that where two forms for Urdu words exist, only the “broken” multiple ligature form is correct. Thus, writing بے کار is the only correct way and بیکار is incorrect. Finally, in the medial Yay should be sorted according to its code, which ever it may be. The code should depend on rules of Urdu diction (which are out of scope of this paper).

Resolution of Issue # 7

The sorting order for the aerab (diacritics) was recommended to be one shown in Figure 5 below. Alif is taken as a representative character to show the aerab, however it should be noted that there are some constraints on placing the aerab. Zer, Zabar, Pesh and Hamza can be placed over all characters except Noon Ghunna. Khari Zabar occurs on vao and Yay, khari Zer is not normally used in Urdu (though included for words borrowed from Arabic) and Ulta Pesh is normally restricted to Gol hay (word finally). In Urdu, Do-Zabar is restricted to Alif (and Do-Zer and Do-Pesh are only included for words borrowed from Arabic). Finally, Tashdeed can occur on all characters (except Alif).



Figure 5: Sorting Sequence of Aerab in Urdu

Figure 6 below shows the recommended sequence for Urdu characters.






ف ق ک کھ گ گ ل لھ م
 مھ ن نھ ن² و وھ ہ ۛ
 ے ی یی ۛ

Figure 6: Sorting Sequence of Urdu Characters³























3.3. Urdu Collation Elements

To realize the given sorting sequence, collation elements for Urdu characters need to be defined. These collation elements are used by the Unicode Collation Algorithm (UCA; equivalent to ISO/IEC 14651) [8, 31]. UCA uses the collation elements to generate sort-keys for the input strings to realize the required lexicographic sorting for Urdu (also see [33, 34] for description of UCA and efficient realization). Collation element is an ordered list of four weights, e.g. 0861.0020.0002.0061 for Latin Small Letter a. The first (left-most) weight is called the *Level 1* weight (or *primary* weight), the second is called the *Level 2* weight (*secondary* weight), the third is called the *Level 3* weight (*tertiary* weight), the fourth is called the *Level 4* weight (*quaternary* weight). These levels are used to distinguish between characters, diacritics, capitalization ('a' vs. 'A') and other lesser significant differences respectively [8]. 0000 sequence is used to make the character ignorable at the level this weight is assigned. Unicode also describes default values for collation elements for all Unicode letters in Allkeys.txt [32]. However, this default sequence cannot realize collation for a specific language. In order to define collation properly for any particular language, these collations codes must be re-ordered in such a way that the coding sequence conforms to the standardized linguistic sorting order. Urdu sorting requires at least three levels. At the first level of sorting, only the basic Urdu characters will be sorted. Once the characters determine word sequence, aerab are used to determine the sequence of words having the same characters. Finally the third level is used to sort special symbols e.g. honorific mark ARABIC SIGN SALLALLAHOU ALAYHE WASSALLAM (see Figure 7 below). Allkeys.txt sequence proposed by Unicode Consortium is used [32] and where required, weights are updated to give the collation sequence required for Urdu.

Glyph	Uni-code	Proposed Collation Elements				English Description
ZWNJ	200C	0000	0010	0002	200C	ZERO WIDTH NON-JOINER
	0600	0000	0000	0000	0600	ARABIC NUMBER SIGN
	0601	0000	0000	0000	0601	ARABIC SIGN SANAH
	0602	0000	0000	0000	0602	ARABIC FOOTNOTE MARKER

² This letter has got a Noon Ghunna Mark (U+0658) on top of it, not currently supported.

³ Do-Chashmee Hay has not been taken as a separate character of Urdu as it is combined with base characters. However, as it is separately encoded in Unicode, it has still been given a collation element, as in Figure 7.

	0603	0000	0000	0000	0603	ARABIC SIGN SAFHA
	0615	0000	0000	0000	0615	ARABIC SMALL HIGH TAH
	060C	0000	0000	0000	060C	ARABIC COMMA
	060D	0000	0000	0000	060D	ARABIC DATE SEPARATOR
	066B	0000	0000	0000	066B	ARABIC DECIMAL SEPARATOR
	066C	0000	0000	0000	066C	ARABIC THOUSANDS SEPARATOR
	061F	0000	0000	0000	061F	ARABIC QUESTION MARK
	061B	0000	0000	0000	061B	ARABIC SEMICOLON
	06D4	0000	0000	0000	06D4	ARABIC FULL STOP
	066A	0000	0000	0000	066A	ARABIC PERCENT SIGN
	060E	0000	0000	0000	060E	ARABIC POETIC VERSE SIGN
	060F	0000	0000	0000	060F	ARABIC SIGN MISRA
	0610	0000	0000	000A	0610	ARABIC SIGN SALLALLAHOU ALAYHE WASSALLAM
	0611	0000	0000	001A	0611	ARABIC SIGN ALAYHE ASSALLAM
	0613	0000	0000	002A	0613	ARABIC SIGN RADI ALLAHOU ANHU
	0612	0000	0000	003A	0612	ARABIC SIGN RAHMATULLAH ALAYHE
	0614	0000	0000	004A	0614	ARABIC SIGN TAKHALLUS
	0652	0000	00C4	0002	0652	ARABIC SUKUN
	064E	0000	00C9	0002	064E	ARABIC FATHA
	0650	0000	00CA	0002	0650	ARABIC KASRA
	064F	0000	00CB	0002	064F	ARABIC DAMMA
	0670	0000	00CD	0002	0670	ARABIC LETTER SUPERSCRIPIT ALEF

	0656	0000	00D5	0002	0656	ARABIC SUBSCRIPT ALEF
	0657	0000	00D8	0002	0657	ARABIC INVERTED DAMMA
	064B	0000	00DB	0002	064B	ARABIC FATHATAN
	064D	0000	00DE	0002	064D	ARABIC KASRATAN
	064C	0000	00E2	0002	064C	ARABIC DAMMATAN
	0654	0000	00E5	0002	0654	ARABIC HAMZA ABOVE
	0651	0000	00E8	0002	0651	ARABIC SHADDA
	0658	0000	00EA	0002	0658	ARABIC MARK NOON GHUNNA
	0653	0000	00F1	0002	0653	ARABIC MADDAH ABOVE
	06F0	0E29	0020	0002	06F0	EXTENDED ARABIC-INDIC DIGIT ZERO
	06F1	0E2A	0020	0002	06F1	EXTENDED ARABIC-INDIC DIGIT ONE
	06F2	0E2B	0020	0002	06F2	EXTENDED ARABIC-INDIC DIGIT TWO
	06F3	0E2C	0020	0002	06F3	EXTENDED ARABIC-INDIC DIGIT THREE
	06F4	0E2D	0020	0002	06F4	EXTENDED ARABIC-INDIC DIGIT FOUR
	06F5	0E2E	0020	0002	06F5	EXTENDED ARABIC-INDIC DIGIT FIVE
	06F6	0E2F	0020	0002	06F6	EXTENDED ARABIC-INDIC DIGIT SIX
	06F7	0E30	0020	0002	06F7	EXTENDED ARABIC-INDIC DIGIT SEVEN
	06F8	0E31	0020	0002	06F8	EXTENDED ARABIC-INDIC DIGIT EIGHT
	06F9	0E32	0020	0002	06F9	EXTENDED ARABIC-INDIC DIGIT NINE
	0627	1350	0020	0002	0627	ARABIC LETTER ALEF
	0622 or 0627 0653	1351	0020	0002	0622	ARABIC LETTER ALEF WITH MADDA ABOVE

ب	0628	1352	0020	0002	0628	ARABIC LETTER BEH
بھ	0628 06BE	1353	0020	0002	0628	ARABIC LETTER BEH + ARABIC LETTER HEH DOCHASHMEE
پ	067E	1354	0020	0002	067E	ARABIC LETTER PEH
پھ	067E 06BE	1355	0020	0002	067E	ARABIC LETTER PEH + ARABIC LETTER HEH DOCHASHMEE
ت	062A	1357	0020	0002	062A	ARABIC LETTER TEH
تھ	062A 06BE	1358	0020	0002	062A	ARABIC LETTER THE + ARABIC LETTER HEH DOCHASHMEE
ط	0679	135A	0020	0002	0679	ARABIC LETTER TTEH
طھ	0679 06BE	135B	0020	0002	062B	ARABIC LETTER TTEH + ARABIC LETTER HEH DOCHASHMEE
ث	062B	135D	0020	0002	062B	ARABIC LETTER THEH
ج	062C	135E	0020	0002	062C	ARABIC LETTER JEEM
جھ	062C 06BE	135F	0020	0002	062C	ARABIC LETTER JEEM + ARABIC LETTER HEH DOCHASHMEE
چ	0686	1361	0020	0002	0686	ARABIC LETTER TCHEH
چھ	0686 06BE	1362	0020	0002	0686	ARABIC LETTER TCHEH + ARABIC LETTER HEH DOCHASHMEE
ح	062D	1364	0020	0002	062D	ARABIC LETTER HAH
خ	062E	1365	0020	0002	062E	ARABIC LETTER KHAH
د	062F	1369	0020	0002	062F	ARABIC LETTER DAL
دھ	062F 06BE	136A	0020	0002	062F	ARABIC LETTER DAL + ARABIC LETTER HEH DOCHASHMEE
ڈ	0688	136B	0020	0002	0688	ARABIC LETTER DDAL
ڈھ	0688 06BE	136C	0020	0002	0688	ARABIC LETTER DDAL + ARABIC LETTER HEH DOCHASHMEE
ذ	0630	1370	0020	0002	0630	ARABIC LETTER THAL

ر	0631	1375	0020	0002	0631	ARABIC LETTER REH
ره	0631 06BE	1376	0020	0002	0631	ARABIC LETTER REH + ARABIC LETTER HEH DOCHASHMEE
رڑ	0691	1377	0020	0002	0691	ARABIC LETTER RREH
رڑھ	0691 06BE	1378	0020	0002	0691	ARABIC LETTER RREH + ARABIC LETTER HEH DOCHASHMEE
ز	0632	137C	0020	0002	0632	ARABIC LETTER ZAIN
زھ	0698	137E	0020	0002	0698	ARABIC LETTER JEH
س	0633	1381	0020	0002	0633	ARABIC LETTER SEEN
ش	0634	1382	0020	0002	0634	ARABIC LETTER SHEEN
ص	0635	1387	0020	0002	0635	ARABIC LETTER SAD
ض	0636	1388	0020	0002	0636	ARABIC LETTER DAD
ط	0637	138C	0020	0002	0637	ARABIC LETTER TAH
ظ	0638	138D	0020	0002	0638	ARABIC LETTER ZAH
ع	0639	138F	0020	0002	0639	ARABIC LETTER AIN
غ	063A	1390	0020	0002	063A	ARABIC LETTER GHAIN
ف	0641	1393	0020	0002	0641	ARABIC LETTER FEH
ق	0642	139B	0020	0002	0642	ARABIC LETTER QAF
ك	06A9	139F	0020	0002	06A9	ARABIC LETTER KEHEH
كھ	06A9 06BE	13A2	0020	0002	06A9	ARABIC LETTER KEHEH + ARABIC LETTER HEH DOCHASHMEE
گ	06AF	13A5	0020	0002	06AF	ARABIC LETTER GAF
گھ	06AF 06BE	13A6	0020	0002	06AF	ARABIC LETTER GAF + ARABIC LETTER HEH DOCHASHMEE

ل	0644	13AB	0020	0002	0644	ARABIC LETTER LAM
لھ	0644 06BE	13AC	0020	0002	0644	ARABIC LETTER LAM + ARABIC LETTER HEH DOCHASHMEE
م	0645	13B0	0020	0002	0645	ARABIC LETTER MEEM
مھ	0645 06BE	13B1	0020	0002	0645	ARABIC LETTER MEEM + ARABIC LETTER HEH DOCHASHMEE
ن	0646	13B4	0020	0002	0646	ARABIC LETTER NOON
نھ	0646 06BE	13B5	0020	0002	0646	ARABIC LETTER NOON + ARABIC LETTER HEH DOCHASHMEE
ں	06BA	13B9	0020	0002	06BA	ARABIC LETTER NOON GHUNNA
ںھ	06BA 06BE	13BA	0020	0002	06BA	ARABIC LETTER NOON GHUNNA + ARABIC LETTER HEH DOCHASHMEE (see Footnote 2)
و	0648	13BD	0020	0002	0648	ARABIC LETTER WAW
وھ	0648 06BE	13BE	0020	0002	0648	ARABIC LETTER WAW + ARABIC LETTER HEH DOCHASHMEE
ہ	06C1	13C2	0020	0002	06C1	ARABIC LETTER HEH GOAL
ھ	06BE	13C4	0020	0002	06BE	ARABIC LETTER HEH DOCHASHMEE
تہ	06C3	13C6	0020	0002	06C3	ARABIC LETTER TEH MARBUTA GOAL
ع	0621	13C7	0020	0002	0621	ARABIC LETTER HAMZA
ی	06CC	13C9	0020	0002	06CC	ARABIC LETTER FARSI YEH
یھ	06CC 06BE	13CB	0020	0002	06CC	ARABIC LETTER FARSI YEH + ARABIC LETTER HEH DOCHASHMEE
ے	06D2	13CE	0020	0002	06D2	ARABIC LETTER YEH BARREE

Figure 7: Collation Elements for Urdu Sorting

While sorting some normalization and/or contractions [8, 31, 33] also need to be done before the collation elements are assigned, in cases where one-to-one mapping is not available between

Urdu character and Unicode encoding, e.g. contraction would collapse two Unicodes ہ and ب

to ہ which is a single character of Urdu. In addition, sometimes there might be instances when

there are multiple ways of encoding the same character(s), e.g. the character لے (Unicode

06D3) may also be written by first writing a لے (Unicode 06D2) and then writing a ے (Unicode

0654).

The collation elements given are sorted by a computer program developed to realize simplified version of four-level sorting (see [8, 31, 33, 34] for details of the algorithm). Figure 8 below shows input and sorted results based on the collation elements.

(a) Input

بھنگی اگنا بیٹی دائرہ گنا عمر گنا آ ب ابھی ایمان ٹیلیفون عمر مان ٹیلی فون گنا اب گنا
بن بہن بی بی عمر دائرہ المعروف آ بن عمر مان بی گنا زکوٰۃ بے زکوٰۃ زکوٰۃ بن
دائرہ بن

(b) Output

اب ابھی ایمان آ آ ب بن بن بن بہن بی بی بی بیٹی بے بھنگی ٹیلی فون ٹیلیفون دائرہ
دائرہ دائرہ المعروف زکوٰۃ زکوٰۃ زکوٰۃ زکوٰۃ عمر عمر عمر گنا گنا گنا گنا گنا
مان مان

Figure 8: Unsorted and Sorted Data for Urdu

The results show that the collation elements can be successfully used to sort for Urdu. However, further work needs to be done to explore if there is need to do further semantic-level Urdu processing for sorting text strings, e.g. in English, sorting of Acronyms "UN", numerals like "9 1/2

weeks" etc. [33]. Further work also needs to be done to define how Arabic script characters in Unicode but not in Urdu (e.g. Sindhi and Pashto characters) would be sorted with Urdu.

4. Conclusion

This paper explains the development and usage of two of the most fundamental standards for Urdu: Urdu character set encoding and Urdu collation sequence. These two standards are essential for the development of Urdu ICTs and would be vital in the generation, exchange and processing of Urdu textual data for ICTs. What has been standardized (or is in the process of being standardized) is not a solution which agrees with all the conventions and norms of Urdu. However, that would be impossible to achieve, as in many case parallel norms exist which are mutually conflicting. Therefore, relevant people and organizations have tried to standardize on best possible reasoning. Nevertheless, natural languages do not always depend on reason and logic and therefore sometimes a level of arbitrariness has to be induced, as has been done here as well. The most important goal and challenge is to preserve the linguistic intuition as much as possible (and not allow the technology to compromise it), though it may not be entirely possible to achieve it totally.

Much more work lies ahead for Urdu to create and mature standards required for ICTs (e.g. Keyboard, locale, etc.) and to develop meaningful applications and content around these standards which can eventually benefit Urdu speakers.

5. Acknowledgements

The historical work discussed here has been possible through inputs from many individuals in their private capacity or acting on behalf of Urdu and Regional Languages' Software Development Forum, National Language Authority, Pakistan Computer Bureau and Ministry of IT & Telecom (formerly Ministry of Science and Technology, IT&T Division).

This work has been partially supported through PAN Localization Project grant from the International Development Research Center, Ottawa, Canada, administered through Center for Research in Urdu Language Processing, National University of Computer and Emerging Sciences, Pakistan.

We would also like to acknowledge Åke Persson, Ernst Tremel, Dr. Khaver Zia and Dr. Muhammad Afzal for their feedback on the paper.

6. References

- [1] **Statement presented by Minister of Information and Communications Technology of Thailand**, at *World Summit for Information Society, Regional Conference for WSIS in Tokyo, Jan 2002*.
- [2] **Rule 5 of the Standard Rules on the Equalization of Opportunities**, UN 1993.
- [3] **Languages of Pakistan**, http://www.ethnologue.com/show_country.asp?name=Pakistan.
- [4] Masood, J. **Pakistan**, in *Digital Review of Asia Pacific 2003/2004*, Orbicom International Secretariat, Montreal, Canada, 2003.
- [5] Hussain, S. and Gul, S. **LICT4D.pk**, accepted for publication in the *Proceedings of Frontiers of IT*, COMSATS Institute of Information Technology, Islamabad, Pakistan, December 2003.
- [6] Hussain, S. **Localization of FOSS**, in the *Proceedings of Free and Open Source Software Asia-Pacific (FOSSAP) Consultation*, APDIP UNDP, Malaysia, February 2004.

- [7] Davis, M. and Durst, M. **Unicode Normalization Forms**, published at <http://www.unicode.org/reports/tr15/tr15-23.html>, Unicode Consortium, 2003.
- [8] Davis, M. and Whistler, K. **Unicode Collation Algorithm**, published at <http://www.unicode.org/reports/tr10/tr10-11.html>, Unicode Consortium, 2004.
- [9] Afzal, M, **Urdu Software Industry: Prospects, Problems and Need for Standards**, *Science Vision* 5(2). Islamabad, Pakistan, 1999.
- [10] Afzal, M. and Hussain, S., **Urdu Computing Standards: Development of Urdu Zabta Takhti**, in the *Proceedings of International IEEE Multi-topic Conference (INMIC)*, Lahore University of Management Sciences (LUMS), Lahore, Pakistan, 2001.
- [11] **Chief Executive Approves Standard Urdu Code Page**, in Akhbar-e-Urdu, National Language Authority, Islamabad, Pakistan, 2000.
- [12] Hussain, S. and Afzal, M. **Urdu Computing Standards: Urdu Zabta Takhti (UZT) 1.01**, in the *Proceedings of International IEEE Multi topic Conference (INMIC)*, Lahore University of Management Sciences (LUMS), Lahore, Pakistan, 2001.
- [13] Unicode Consortium, *Unicode 4.0*, Addison-Wesley, Reading, UK, 2003.
- [14] Zia, K., **Towards Unicode Standard for Urdu**, in the *Proceedings of Fourth Symposium on Multi-lingual Information Processing (MLIT4)*, Yangoon, Myanmar (CICC Japan), 1999.
- [15] Hussain, S., **Proposal to Add Marks and Digits in Arabic Code Block (for Urdu)**, in the *Proceedings of ISO/IEC JTC1/SC2/WG2 Meeting*, Dublin, Ireland, 2002.
- [16] Bhurgari, A. M., **Enabling Pakistani Languages through Unicode**, published at <http://download.microsoft.com/download/1/4/2/142aef9f-1a74-4a24-b1f4-782d48d41a6d/PakLang.pdf>.
- [17] Wissink, C. and Kaplan, M., **Sorting it all out: An Introduction to Collation**, in the *Proceedings of 23rd International Unicode Conference*, March 2003, Prague, Czech Republic.
- [18] Hussain, S. and Karamat, N. **Urdu Collation Sequence**, in the *Proceedings of International IEEE Multi topic Conference (INMIC)*, Ghulam Ishaq Khan Institute (GIKI), Islamabad, Pakistan, 2003.
- [19] *Feroz-ul-Lughat Jamay (فیروز اللغات جامع)*, Feroz Sons, Lahore, Pakistan.
- [20] *Standard Twentieth Century Dictionary: Urdu to English*, Educational Publishing House, New Dehli, India.
- [21] Haqqi, S., *Farhang-e-Talafuz (فرہنگ تلفظ)*, Muqtadra Qaumi Zaban, Umaiz Publications, 1995, Islamabad, Pakistan.
- [22] *Jadeed Urdu Lulghat (جدید اردو لغت)*, Muqtadra Qaumi Zuban, Islamabad, Pakistan.
- [23] *Urdu Lughat (اردو لغت)*, Urdu Lughat Board, Karachi, Pakistan.

[24] Platts, J., *A Dictionary of Urdu, Classical Hindi and English*, Crosby Lockwood and Son, London, UK, 1911, reprinted by Sang-e-Meel Publishers, Lahore, Pakistan.

[25] Dehlvi, M. S. A., *Farhang-e-Asfia* (فرہنگ آصفیہ), reprinted by Sang-e-Meel Publications, Lahore, Pakistan, 2002.

[26] Nayyer, M. N. H., *Noor-ul-Lughat* (نورالغٹ), Sang-e-Meel Publications, 1989, Lahore, Pakistan.

[27] Platts, J., *A Grammar of the Hindustani or Urdu Language*, 1909, reprinted by Sang-e-Meel Publishers, Lahore, Pakistan.

[28] *Almi Urdu Lughat*.

[29] *Naseem-ul-Lughat*.

[30] Hussain, S., **Collation Sequence of Pakistani Languages**, invited talk at *Seminar on Software Development of Urdu and Other Pakistani Languages* held at Karachi Institute of Information Technology, September 2002.

[31] **ISO/IEC 14651: Information Technology - International String Ordering and Comparison - Method for Comparing Character Strings and Description of Common Template Tailorable Ordering**, ISO Geneva, Switzerland (www.iso.ch), 2001.

[32] Unicode Consortium, **Default Unicode Collation Element Table**, <http://www.unicode.org/Public/UCA/latest/allkeys.txt>, 2003.

[33] Gillman, R., *Unicode Demystified: A Practical Programmer's Guide to the Encoding Standard*, Addison-Wesley, Boston, USA, 2003.

[34] International Components for Unicode, **IBM, ICU Collation Design Documentation**, http://oss.software.ibm.com/cvs/icu/icuhtml/design/collation/ICU_collation_design.htm.