International Journal of Engineering Research and Development
eISSN : 2278-067X, pISSN : 2278-800X, www.ijerd.com
Volume 4, Issue 8 (November 2012), PP. 101-108

Compression Technique based on Dictionary approach for Gujarati Text

Sandip V. Maniya[1], Ms. Jikitsha Sheth [2], Dr. Kalpesh Lad [3]
[1]*Shrimad Rajchandra Institute of Management and Computer Application, Uka Tarsadia university, MalibaCampusGopal Vidyanagar, Bardoli MahuvaRoad, Tarsadi, Dist: Surat-394350, Gujarat (INDIA).*

Abstract:—There are number of data compression algorithms, which are dedicated to compress English text. This paper proposed a new data compression and decompression method for Gujarati text. The proposed technique is also compared with Huffman compression coding. Procedure prescribed in this paper compresses the Gujarati text to 60%; it is Unicode base lossless compression technique.
Keywords:—Data compression, Lossless compression, Lossy compression, UTF-16, UTF-8, Huffman coding, Gujarati text compression, Dictionary, Unique index number, Compression ratio.

I. INTRODUCTION

Data compression is the process of encoding the data, so that fewer bits will be needed to represent the original data whereby the size of the data is reduced [1]. Encoded data must be decoded in order to make use of them. It reduces storage cost by storing more logical data per unit of physical capacity; it demonstrates an accurate understanding of the properties of the data, which is important for applications such as data warehousing, telecommunications, cryptography, speech recognition and spell checking [2]-[3].

Compression process is classified into lossless compression and lossy compression [4]. Lossless algorithms do not change the content of a file; if we compress a file and then decompress it, it has not changed, but in lossy compression if we compress a file and then decompress it, it has changed [4]. Image, audio and video file converts into larger to smaller file through some method; number of pixel is decrease so resolution is poor and data is loss then original data, so they are suitable for processing audio and video files where loss of resolution is ignored, depending on the preferred quality [5].
The next billion Internet users will come mostly from Asia; but there is very little content in local languages to support them, Just 13.83% of all content on the Internet was in Asian languages [18]. And from that only 0.03% web contents accounted other than Chinese, Japanese and Korean Asian languages including Gujarati [18]. According to data, Gujarati text is very less available in electronic form, many compression techniques have been proposed for English text [5].

In this paper we examine techniques for compressing Gujarati text, considering both special-purpose methods designed for Gujarati and general-purpose compression tools. To establish a new benchmark for compression of Gujarati, we investigated in detail the most effective general-purpose compression technique such as Shannon-Fano coding, Huffman, LZW, RLE and testing several refinements to improve its performance on large alphabet languages such as Gujarati. In particular we show that the standard techniques for escape prediction and memory management do not work well for Gujarati, and propose modifications that cater to the characteristics of Gujarati text. In this paper authors proposed a new compression method which is work on Unicode and lossless for Gujarati text.

II. UNICODE

An early important example of fixed-length binary coding for text was the well known 5-bit Murray Teletype code; Early computers quickly moved to various proprietary 6-bit codes and then, in the early 1960's, to 7-bit ASCII and 8-bit EBCDIC, these becoming the standard text representations for the next quarter century [6]-[7].

Unicode is a Universal character set table that contains 65,535 characters that cover almost all the characters, punctuations, and symbols in the world. The Unicode coded character set is the largest of its kind. Almost a million code positions are available in Unicode for formal character encoding, with more than 137,000 additional code positions reserved for private-use characters [6]. The "canonical" Unicode representation is the 16-bit UCS-2. The UTF-8 recoding allows ASCII characters to be represented in 8 bits, but expands others to 2 or 3 bytes and is often used as a distribution format.

UTF-16 is not suitable for all applications, ASCII-based operating systems and text-processing software. Not only did UTF-16 use 16 bits for all characters, but all bytes are possible [6]. To solve these problems and allow the adoption of Unicode in these situations, another encoding form, UTF-8, was devised. UTF-8 encodes all ASCII characters using the same bytes as in ASCII, and does not use those bytes to encode anything else; Today UTF-8 is by far the most common Unicode text format [7]. Here we use the UTF-8 base Gujarati text compression and compare Huffman compression with new proposed compression technique.

III. CHARACTERISTICS OF GUJARATI TEXT

There are about 65.5 million speakers of Gujarati worldwide, making it the 26th most spoken native language in the world [16]. Gujarati is one of the 22 official languages and 14 regional languages of India [17]. Though it is a very widely spoken language, limited work is found in the literature that addresses the recognition of Gujarati language. Like

other languages Sanskrit, Hindi, Marathi which has been derived from Devanagari, some of the Guajarati characters are very similar in appearance. The major difference between English and Gujarati is that the latter has a much larger set of characters. There are two main difference between Gujarati Text and English text compression, 1) the compression techniques involving pseudo-coding of letters are not applicable for Gujarati text and 2) we may employ specific mechanism of coding dependent vowel signs to remove redundancy in Gujarati that is absent in case of English.

Though English has got a fixed encoding base long ago, still now in practical applications, Gujarati has not adapted unique encoding scheme. The use of Gujarati Unicode has not yet got a massive use. This is really a great limitation for research in Gujarati. Gujarati text compression also suffers from the same problem.

IV. HUFFMAN CODING WITH GUJARATI TEXT

Compression theory distinguishes two activities: the construction of a model that associates a probability with each symbol; and coding to produce a compressed representation of data with respect to the probability of each symbol [8].

The Huffman code algorithm creates variable length code that is integral number of bits. The Huffman code have unique prefix codeword for symbol base on symbol probability distribution pi, where I = 1, 2, 3. . . N; the frequency distribution of all the symbols of the source is calculated in order to calculate the probability distribution. According to the probabilities, the codeword for each symbol are assigned. It assigns shorter codeword for higher probability symbols and longer codeword for smaller probability symbols. [9]-[11].The Huffman codes are built from the bottom up, starting with the leaves of the tree and working progressively closer to root [10].

Building Huffman decoding tree is done using completely different Huffman code is laid out as string of leaf nodes that are going to be connected by a binary tree. Each node has weight, which is simply occurrence or probability of symbol's appearance [8]. Huffman may or may not have had digital computer in mind when he developed his code, but programmer use the tree structure all the time. [8]. Following example describe the Huffman coding algorithm in tree structure.

Input String: મારુ નામ સંદિપ છે.

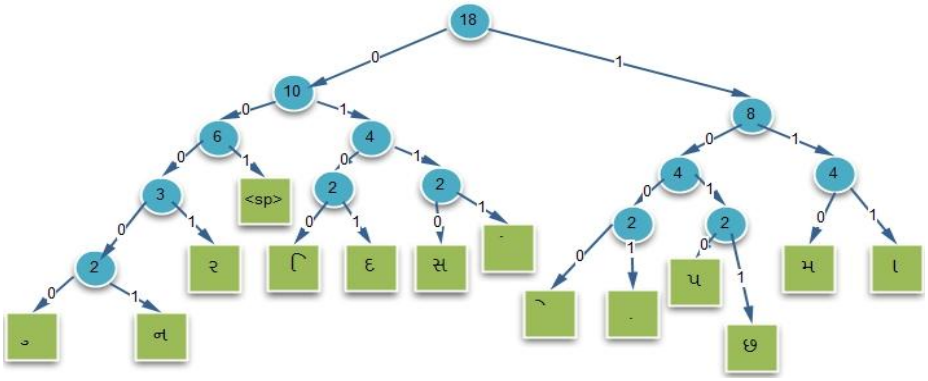


Figure 1.1 *Huffman tree*

Encoding:

Input: મારુ નામ સંદિપ છે.

Encoding of "મારુ નામ સંદિપ છે." string using Huffman code generates the following bits stream.

મ	ા	ર	ુ	space	ન	ા	મ	space	સ	ં	ઇ	ૅ	પ	space	છ	ે	.
110	111	0001	00000	001	00001	111	110	001	0110	0111	0101	0100	1010	001	1011	1000	1001

Output: 1101110001000000010000111111000101100111010101001010001101110001001

Decoding:

Input: 1101110001000000010000111111000100001011101010100101000100110111000 1001

At time of decompression decode the encoded bit stream. So put the original character according to code word and retrieve the original string.

110	111	0001	00000	001	00001	111	110	001	00001	0111	0101	0100	1010	001	1011	1000	1001
મ	ા	ર	ુ	space	ન	ા	મ	space	સ	ં	ઇ	ૅ	પ	space	છ	ે	.

Output: મારુ નામ સંદિપ છે.

V. PROPOSED APPROACH FOR GUJARATI TEXT COMPRESSION

In this paper, we propose a new dictionary for Gujarati text compression. To facilitate efficient searching of the text, we employ term unique compression code to each and every character of Gujarati language dictionary entries. Dictionary of Gujarati characters can be found in Annexure-I. This dictionary includes 78 Gujarati character with general Unicode and English keyboard characters. Extract two characters at a time from the file and once the dictionary is created proceed through the following steps until end of the file is reached.

Extract Unique Index Number and Length of Unique Index Number from dictionary. (if Extract 'ખ' than it Unique Index Number is 42 in dictionary and length of it is 2).

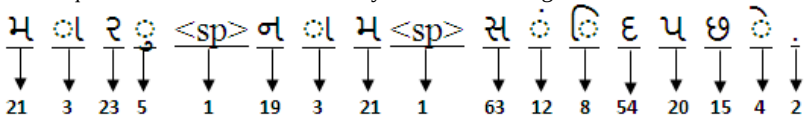
If 1st character length is 2 and 2nd character length is 1 than both characters unique index number are concatenated (such as, 1st character unique index number is 42 and other is 4 than concatenation is 424) otherwise unique index number remain as it is.

Concatenated number or Unique Index number converts into related to original Unicode character.

In this method most probability two Gujarati characters become in single character so many bits are reduce. All Gujarati character Unicode size is maximum 24 bits. So this method is very useful to reduce bit size of Gujarati text.

Let us take a "મારુ નામ સંદિપ છે." string and apply proposed compression method.

Get unique numbers form the dictionary for each of string is as below.



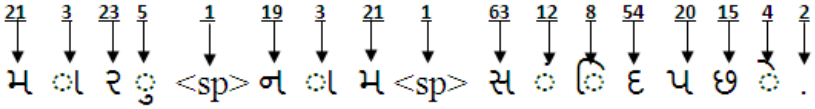
Combinations of unique numbers are as below.

213 235 119 321 63 128 54 20 154 2

Encoding of above combinations numbers into Unicode is final compressed code. Decoding is reverse process, for that get the Unicode character from dictionary then convert into Unicode number.

213 235 119 321 63 128 54 20 154 2

If length of is 3 than disjoint it.



So we can get original string is as below.

મારુ નામ સંદિપ છે.

VI. RESULT AND DISCUSSION

Authors implement proposed method in C++ language and compression different Gujarati content text file. Also same file compressed with the Huffman coding and based on that generate the following table.

Table I: Gujarati text file size compression data

Text File	**Original File size**	**Huffman Coding**		**Proposed Algorithm**	
		Output File	**Compression Ratio**	**Output File**	**Compression Ratio**
Test1.txt	21.7 KB	10.4 KB	52.07 %	8.55 KB	60.59 %
Test2.txt	149 KB	70 KB	53.02 %	58.9 KB	60.46 %
Test3.txt	1,425 KB	663 KB	53.47 %	560 KB	60.70 %
Test4.txt	2,778 KB	1,294 KB	53.41 %	1,094 KB	60.61 %
Test5.txt	5,387 KB	2,510 KB	53.40 %	2,121 KB	60.62 %
Test6.txt	10,864 KB	5,061 KB	53.41 %	4,276 KB	60.64 %
Test7.txt	27055 KB	12,601 KB	53.42 %	10,649 KB	60.63 %
Test8.txt	30,812 KB	14,351 KB	53.42 %	12,127 KB	60.64 %
Average compression Ratio:			53.20 %		60.61 %

Authors create different Gujarati text files size on disk and compressed them using Huffman coding and proposed compression method. Based on that generates the following table.

Table III: Gujarati text file size on disk compression data

Text File	Original File size on disk	Huffman Coding		Proposed Algorithm	
		Output File size on disk	Compression Ratio	Output File size on disk	Compression Ratio
Test1.txt	24 KB	12 KB	50 %	12 KB	50 %
Test2.txt	152 KB	72 KB	52.63 %	60 KB	60.52 %
Test3.txt	1,428 KB	664 KB	53.50 %	564 KB	60.50 %
Test4.txt	2,780 KB	1,296 KB	53.38 %	1,096 KB	60.57 %
Test5.txt	5,388 KB	2,512 KB	53.37 %	2,124 KB	60.57 %
Test6.txt	10,864 KB	5,064 KB	53.38 %	4,276 KB	60.64 %
Test7.txt	27,056 KB	12,604 KB	53.41 %	10,652 KB	60.62 %
Test8.txt	30,812 KB	14,352 KB	53.42 %	12,128 KB	60.63 %
Average compression Ratio:			52.88 %		59.25 %

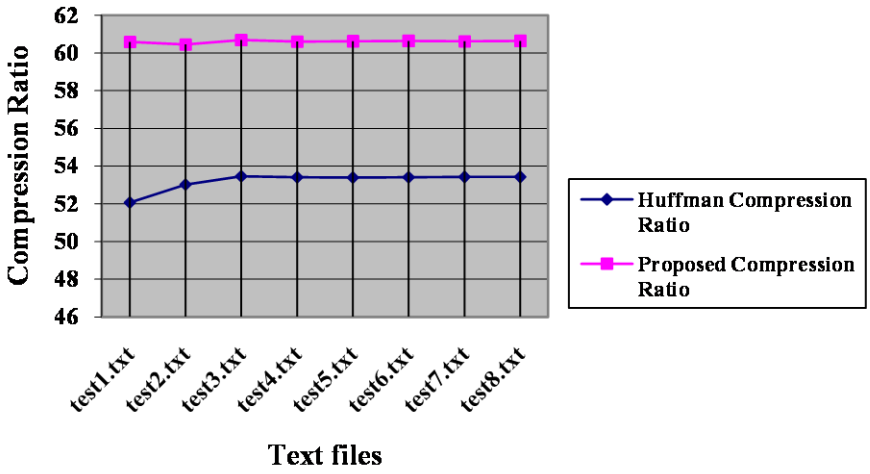
In above first case, authors compress a Gujarati text file and focus on file size. The compression ratio as a measure of efficiency has been considered and can be calculated as,

Compression ratio= ((compression size*100)/Source file size) % (1)

Compression size is

Compression size= Input Original file size-Output Compressed file size (2)

As per table I Gujarati text on file size compression data, Graph is as below.



Here we show that average Huffman code compression ratio of file size is 53.20 % and average proposed compression ratio of file size is 60.61 % so that is clearly identifying that file size difference is approximately 7.41 % more compression by proposed compression method than Huffman coding compression algorithm.

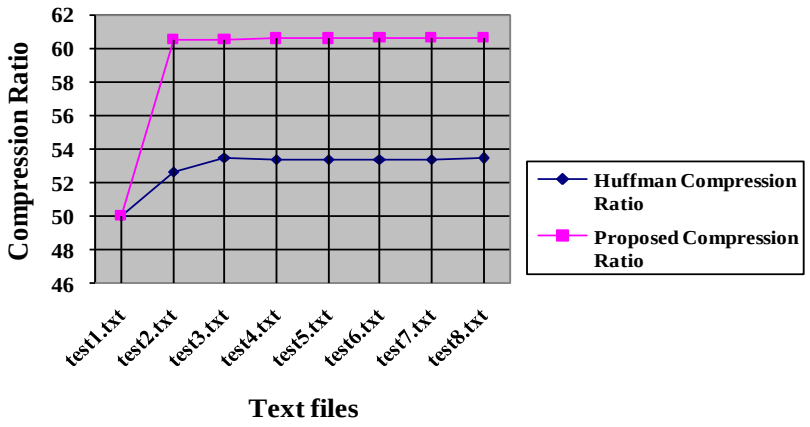
In second case, Gujarati text compression files with size on disk. The compression ratio as a measure of efficiency has been considered and can be calculated as,

Compression ratio= ((compression size on disk*100)/Source file size on disk) % (1)

Compression size on disk is

Compression size on disk= Input Original file size on disk-Output Compression file size on disk (2)

As per table II Gujarati text file size on disk compression data, Graph is as below.



Here we show that average Huffman coding compression ratio with file size on disk is 52.88 % and average proposed compression method ratio with file size on disk is 59.25 % so that is clearly identifying that file size on disk difference is approximately more than 6.37 % compression by proposed method than Huffman coding compression algorithm.

VII. CONCLUSION

The proposed method is one of the initiating steps of Enhanced Text Representation Scheme for Gujarati Text. In this step, a novel approach of constructing data compression dictionary has been proposed which is also an innovative approach of Gujarati text compression. We have impressive outcomes of the proposed approach in terms of compression size and compression ratio. Also observe that proposed method gives better result compare to Huffman coding compression method for Gujarati text.

As the proposed method is adapted for both conventional encoding and Unicode standard, it may be employed very easily for any Gujarati text compression. Communication of Gujarati Small Text Message may also be immensely facilitated with the presented approach of Gujarati text compression. The proposed method is also to some extent an initialization of Gujarati text compression approaches.

REFERENCES

[1]. Raveen V, "Through the History and Mystery of Data Compression", Cover Story, CSI Communications, March-2012, PP. 6-8.

[2]. Ester M., Kriegel H., Sander J., Xu X. "A lossless based Algorithm for compressing Clusters in Large Databases with audio data and noise", Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, 1996, PP. 226-231.

[3]. M.K. Sharma, Jyotsana Sah, "Applications of Data Compression Approach In Data Warehouse Design", Proceeding of COIT-2008, RIMT-IET, Mandi Gobindgarh, March- 2008, PP. 31-33.

[4]. S.R. Kodituwakku, U. S.Amarasinghe, "comparison of lossless data compression algorithms for text data", Indian journal of computer science and engineering, vol 1, no 4, ISSN: 0976-5166, pp. 416-425.

[5]. S Sankar, Dr. S Nagarajan, "A Comparative Study: Data Compression on TANGLISH Natural Language Text", International Journal of Computer Applications, ISSN: 0975–8887, Vol. 38– No.3, January 2012 33, PP. 33-37.

[6]. Doug Ewell, "A survey of Unicode compression", January-2004, www.unicode.org.

[7]. Atkin, Steven and Stansifer, Ryan. "Unicode Compression: Does Size Really Matter?", presented at the 22nd International Unicode Conference, San José, Calif., September 2002.

[8]. David Salomon, "Data Compression-The Complete Reference Book", 4th Edition, Springer.

[9]. Mark Nelson & Jean-loup Gailly, "The Data Compression Book", 2nd edition. M & T Books, New York, ISBN 1-55851-434-1, 1996.

[10]. David Huffman, "A method for the construction of minimum redundancy codes", Proceeding of IRE, Vol. 40, No. 9, September 1952,PP. 1098-1101.

[11]. T.C. Bell, J.G. Cleary, and I.H. Witten, "Text Compression", Prentice-Hall, Englewood, New Jersey, 1990.

[12]. J. N. Chen, J. S. Chang and M.H. Chen. "Using Word-Segmentation Model for Compression of Chinese Text", Computer Processing of Chinese and Oriental Languages, pp. 17-30, 1995.

[13]. S. A. Ahsan Rajon and Md. Rafiqul Islam, "An Effective Approach for Compression of Bengali Text", IJCIT, ISSN 2078-5828, Vol. 01, Issue 02, PP. 30-36.

[14]. F. Awan and A. Mukherjee, "LIPT: A Lossless Text Transform to improve compression", Proceedings of International Conference on Information and Theory: Coding and Computing, IEEE Computer Society, Las Vegas Nevada, 2001.
[15]. Mohd. Faisal Muqtida Raju Singh Kushwaha," Improvement in Compression Efficiency of Huffman Coding", International Journal of Computer Applications, ISSN 0975 – 8887, Volume 45– No.24, May 2012,PP. 33-39.
[16]. Gujarati Language Facts and Information, Article, www.phrasebase.com/languages/gujarati/ visited on 25th September 2012.
[17]. Babu Suthar, "Gujarati-English learner Dictionary", A Nirman Foundation Project, Department of South Asia Studies, University of Pennsylvania, Philadelphia 2003.
[18]. Local Language Asian Portals and content, www.asiaonline.net visited on 25th September 2012.

Annexure-I

U_NO	UNICODE CHARACTER MENING	UNI_CHAR	DECIMAL
0	GUJARATI SIGN CANDRABINDU	ઁ	2689
1	GUJARATI SIGN ANUSVARA	ં	2690
2	GUJARATI SIGN VISARGA	ઃ	2691
3	GUJARATI LETTER A	અ	2693
4	GUJARATI LETTER AA	આ	2694
5	GUJARATI LETTER I	ઇ	2695
6	GUJARATI LETTER II	ઈ	2696
8	GUJARATI LETTER U	ઉ	2697
9	GUJARATI LETTER UU	ઊ	2698
11	GUJARATI LETTER VOCALIC R	ઋ	2699
12	GUJARATI VOWEL CANDRA E	ઍ	2701
14	GUJARATI LETTER E	એ	2703
15	GUJARATI LETTER AI	ઐ	2704
16	GUJARATI VOWEL CANDRA O	ઑ	2705
17	GUJARATI LETTER O	ઓ	2707
18	GUJARATI LETTER AU	ઔ	2708
19	GUJARATI LETTER KA	ક	2709
20	GUJARATI LETTER KHA	ખ	2710
21	GUJARATI LETTER GA	ગ	2711
22	GUJARATI LETTER GHA	ઘ	2712
23	GUJARATI LETTER NGA	ઙ	2713
24	GUJARATI LETTER CA	ચ	2714
25	GUJARATI LETTER CHA	છ	2715
27	GUJARATI LETTER JA	જ	2716
28	GUJARATI LETTER JHA	ઝ	2717
29	GUJARATI LETTER NYA	ઞ	2718
30	GUJARATI LETTER TTA	ટ	2719
31	GUJARATI LETTER TTHA	ઠ	2720
32	GUJARATI LETTER DDA	ડ	2721
33	GUJARATI LETTER DDHA	ઢ	2722
34	GUJARATI LETTER NNA	ણ	2723

35	GUJARATI LETTER TA	ત	2724
36	GUJARATI LETTER THA	થ	2725
37	GUJARATI LETTER DA	દ	2726
38	GUJARATI LETTER DHA	ધ	2727
39	GUJARATI LETTER NA	ન	2728
40	GUJARATI LETTER PA	પ	2730
41	GUJARATI LETTER PHA	ફ	2731
42	GUJARATI LETTER BA	બ	2732
43	GUJARATI LETTER BHA	ભ	2733
44	GUJARATI LETTER MA	મ	2734
45	GUJARATI LETTER YA	ય	2735
46	GUJARATI LETTER RA	ર	2736
47	GUJARATI LETTER LA	લ	2738
48	GUJARATI LETTER LLA	ળ	2739
49	GUJARATI LETTER VA	વ	2741
50	GUJARATI LETTER SHA	શ	2742
50	GUJARATI LETTER SSA	ષ	2743
52	GUJARATI LETTER SA	સ	2744
53	GUJARATI LETTER HA	હ	2745
54	GUJARATI SIGN NUKTA	઼	2748
55	GUJARATI SIGN AVAGRAHA	ઽ	2749
56	GUJARATI VOWEL SIGN AA	ા	2750
57	GUJARATI VOWEL SIGN I	િ	2751
58	GUJARATI VOWEL SIGN II	ી	2752
59	GUJARATI VOWEL SIGN U	ુ	2753
60	GUJARATI VOWEL SIGN UU	ૂ	2754
61	GUJARATI VOWEL SIGN VOCALIC R	ૃ	2755
62	GUJARATI VOWEL SIGN VOCALIC RR	ૄ	2756
63	GUJARATI VOWEL SIGN CANDRA E	ૅ	2757
64	GUJARATI VOWEL SIGN E	ે	2759
65	GUJARATI VOWEL SIGN AI	ૈ	2760
66	GUJARATI VOWEL SIGN CANDRA O	ૉ	2761
67	GUJARATI VOWEL SIGN O	ો	2763
68	GUJARATI VOWEL SIGN AU	ૌ	2764
69	GUJARATI SIGN VIRAMA	્	2765
70	GUJARATI OM	ૐ	2768
71	GUJARATI LETTER VOCALIC RR	ૠ	2784
72	GUJARATI DIGIT ZERO	૦	2790
73	GUJARATI DIGIT ONE	૧	2791
74	GUJARATI DIGIT TWO	૨	2792

75	GUJARATI DIGIT THREE	૩	2793
76	GUJARATI DIGIT FOUR	૪	2794
77	GUJARATI DIGIT FIVE	૫	2795
78	GUJARATI DIGIT SIX	૬	2796
79	GUJARATI DIGIT SEVEN	૭	2797
80	GUJARATI DIGIT EIGHT	૮	2798
81	GUJARATI DIGIT NINE	૯	2799
82	<ENTER>	<ENTER>	10
83	<SPACE>	<SPACE>	32
84	FULL STOP	.	46
85	<TAB>	<TAB>	9
86	EXCLAMATION MARK	!	33
87	QUOTATION MARK	“	34
88	NUMBER SIGN	#	35
89	DOLLAR SIGN	$	36
90	PERCENT SIGN	%	37
91	AMPERSAND	&	38
92	APOSTROPHE	‘	39
93	LEFT PARENTHESIS	(	40
94	RIGHT PARENTHESIS	)	41
95	ASTERISK	*	42
96	PLUS SIGN	+	43
97	COMMA	,	44
98	HYPHEN-MINUS	-	45
99	SOLIDUS	/	47
1024	DIGIT ZERO	0	48
1025	DIGIT ONE	1	49
1026	DIGIT TWO	2	50
1027	DIGIT THREE	3	51
1028	DIGIT FOUR	4	52
1029	DIGIT FIVE	5	53
1030	DIGIT SIX	6	54
1031	DIGIT SEVEN	7	55
1032	DIGIT EIGHT	8	56
1033	DIGIT NINE	9	57
1034	COLON	:	58
1035	SEMICOLON	:	59
1036	LESS-THAN SIGN	<	60
1037	EQUALS SIGN	=	61
1038	GREATER-THAN SIGN	>	62
1039	QUESTION MARK	?	63
1040	COMMERCIAL AT	@	64
1041	LEFT SQUARE BRACKET	[	91
1042	REVERSE SOLIDUS	\	92
1043	RIGHT SQUARE BRACKET	]	93
1044	CIRCUMFLEX ACCENT	^	94
1045	LOW LINE	_	95
1046	GRAVE ACCENT	`	96
1047	LEFT CURLY BRACKET	{	123
1048	VERTICAL LINE	|	124
1049	RIGHT CURLY BRACKET	}	125
1050	TILDE	~	126