Sindh Univ. Res. Jour. (Sci. Ser.) Vol.43 (1) 43-50 (2011)

SINDH UNIVERSITY RESEARCH JOURNAL (SCIENCE SERIES)

Automatic Diacritics Restoration System for Sindhi

J. A. MAHAR AND G. Q. MEMON

Faculty of Engineering, Science and Technology, Hamdard University, Karachi, Pakistan

Corresponding author: J. A. MAHAR, E-mails: mahar.javed@gmail.com, Ph: +92-334-2727937.

Received 28rth March 2011 and Revised 10th May 2011)

Abstract: Sindhi language is based on the pattern of Arabic script and usually both are written without diacritics in the routine applications. The absence of diacritics creates many ambiguities and confusions for the possible vowel sounds of the group of characters used in the composition of the word. Moreover, the morphological and lexical ambiguity is also a case for the correct pronunciation in computational systems. Realizing the cause, this paper is composed to present an innovated and improved mechanism that inserts the diacritic signs correctly into the non-diacritized text by the multiplications of three N-gram probabilities with Viterbi algorithm, the probabilities of words are calculated by using unigram, bigram and trigram models. The performance of system is achieved in word error rate as 0.71% and diacritic error rate as 3.21%. A few languages i.e., Arabic, Urdu and Persian have the same characteristics as Sindhi does for the reason proposed system may be useful for mentioned languages on same scale.

Keywords: N-grams; Viterbi Algorithm; Diacritics Restoration; Sindhi Language

INTRODUCTION

Most of the words found in Arabic, Persian, Urdu and Sindhi orthographically look similar but are different in their meanings because the orthographic system of these languages use superscript and subscript diacritical marks unlike in Sindhi routine writings are devoid of such diacritics. However, diacritics are placed only where there is inadequate contexts. The purpose of this research is to develop an automatic system that converts non-diacritic into diacritized text. Therefore, this improved mechanism of diacritic restoration is proposed that is based on the multiplication of three N-grams i.e., unigram, bigram and trigram. The N-grams are probabilistic models which provide direction to assign probabilities to words; it represents an n^{th} order Markov language model (Jurafsky, *et al*., 2000). The viterbi algorithm is used to select the highest probability diacritized word from the given list of homographic words.

The statistical language modeling is an ever challenging task for morphologically rich languages and always requires the corpus of the language. The corpus of Sindhi language is selected for diacritics restoration because many letters of Arabic, Persian and Urdu are included in Sindhi alphabet and it has a large number of homographic words in its particular texts.

The widespread usage of computers in linguistic applications; Sindhi texts need to be supplied with diacritics in order to be correctly vocalized before being processed. Thus the application of automatic diacritic restoration for Sindhi computational processing is as important as the life for a language. The automatic insertion of diacritics into written Sindhi is important for various natural language processing applications, including search engines, text to speech engines, speech recognition, text mining, information retrieval, machine translation, mobile message reading, talking dictionaries and corpora acquisition. Relatively they are also useful for personal, official, industrial uses as well as learners of language.

The diacritics restoration is a lexical disambiguation task in natural language processing, to an extent this problem has been solved by various researchers using word-based, grapheme-based and character-based approaches; each approach has qualities and deficiencies with respect to different natures of languages. These approaches are based on various developed or adopted techniques for the task of diacritics restoration for instance, hidden markov model is used by (Gal, 2002), (Harby, *et al*., 2008) and (Elshafei, *et al*., 2006), the maximum entropy is proposed by (Zitouni, *et al*., 2006), the application of neural networks is investigated by (Sultan, 2001), the weighted finite state

transducers are proposed by (Nelken, *et al.,* 2005) and (Rashwan, *et al*., 2011) introduced a system that is based on two-layer stochastic. The linguistic knowledge and lexical resources are also used for diacritics restoration for instance, the morphology and other linguistic features are investigated by (Roth, *et al*., 2008) and lexical resources are used by (Habash, *et al*., 2007), the Word Net approach is proposed by (Mahar, *et al*., 2011). The acceptable results have been achieved for various European and African languages but Arabic script based languages still need more attention due to rich morphological structure.

MATERIAL AND METHODS

2.1 Material

The corpus of language is necessary for diacritics restoration using statistical approaches therefore, شاه جو رسالو (Shah Jo Risalo) is selected as corpus L because in this book original Sindhi language is used by the great poet Shah Abdul Latif Bhitai. The lexicon LSJR containing 27360 words is developed for training and testing dataset. The Shah Jo Risalo managed by (Aadwani, K., 2009) is used which is divided into 30 tune chapters, each chapter is based on poems and lays. The approximate length of the poems given from 2 to 11 lines, the total number of poems is 1579 the length of lays is given from 4 to 16 lines approximately the total number of lays is 43. The statistical information of words in Shah Jo Risalo is shown in (**Table 1**).

Table 1: Words information of Shah Jo Risalo

Total No. of Word	27360
Total No. of Word Types	10894
Total No. of un-ambiguous Words	9085
Total No. of ambiguous Words	1809

2.2 Methods

The diacritic restoration system compares the structure of non-diacritized word with diacritized word and then picks the diacritized word with the highest count. The N-gram models and Viterbi algorithm are well known and available in literature (Jurasfky, *et al*., 2000), the contribution in this research is to perform multiplication of three N-grams i.e., unigram, bigram and trigram for calculating the highest probability values of sequence of words, on the bases of calculated probability, Viterbi algorithm is used for finding the most likely path. The short overview of N-gram models and Viterbi algorithm is given below:

The text is a sequence of words that can be represented as: $W_1, W_{2,\ldots,} W_n$ or W_1^n. If each word is occurring in its appropriate position then probability can be represented as:

$$P(W_1, W_2, \ldots W_{n-1}, W_n) \qquad (1)$$

The bigram model is a first-order Markov model because it looks for one word from the past that can be generalized to the trigram (second-order) because it looks for two words from the past

$$P(w_i \mid w_{i-1}) \qquad (2)$$

$$P(w_i \mid w_{i-2} w_{i-1}) \qquad (3)$$

The diacritic bigram for the process of diacritic restoration is similar to (Harby, *et. al*., 2008) except that we only calculate the hidden states. The probability for sequence of diacritic marks (hidden states) T_1, T_2, T_N is defined as:

$$\sim \prod_{i>1}^{n} P(T_i \mid T_{i-1}) \qquad (4)$$

The methodology of our proposed system is mainly based on the multiplication of calculated probabilities of unigram, bigram and trigram models therefore, process of Sindhi diacritic restoration system is defined as:

$$P(T_i \mid T_{i-1}) P(W) P(W_i \mid W_{i-1}) P(W_i \mid W_{i-2} W_{i-1}) \qquad (5)$$

The Viterbi algorithm is best described in (Harby, *et al*., 2008). We have modified Viterbi algorithm according to nature of the task of diacritics restoration. The modified algorithm is written as under:

1 Initialization Step

For i = 1 to N
{
k = 1
DATA (i, k) =
$P(T_i \mid T_{i-1}) P(W) P(W_i \mid W_{i-1}) P(W_i \mid W_{i-2} W_{i-1})$
k = k + 1
}

2 Iteration Step

For j = 1 to N
{
LOC = 1
MAX = DATA (j, 1)
M = LENGTH (DATA (j, k))
For k = 2 to M
{
If (MAX < DATA (j, k) then
{
MAX = DATA (j, k)
LOC = k
}
PTR (j) = MAX
}
}

3 Sequence Display Step
For p = 1 to N
Write PTR (p)

3. Implementations

The semantic analysis is difficult for solving lexical ambiguities, therefore identification of correct contextual relationship is focused, in this regard, the multiplication of three N-gram models i.e., unigram, bigram and trigram with Viterbi algorithm is used for diacritic restoration system of Sindhi language. The unigram, bigram and trigram models are trained on corpus of diacritized text of Shah Jo Risalo and then tested with non-diacritized text of same corpus. This section presents the implementation process of tokenization, probability calculations and Viterbi algorithm.

3.1 Tokenization

The tokenization is the first process of this proposed mechanism; it is the process of segmenting input sequence of orthographic symbols, the division of input text into tokens is necessary for language modeling. As the orthography of Sindhi is based on the concatenation of syllables and words are normally delimited through white spaces like in English, but segmentation of words sometimes may be ambiguous due to the presence of embedded space in a single word, for instance, a word صاحب قدرت is a compound word and we explicitly put hard space between صاحب and قدرت , this explicit hard space is affecting on tokenization, therefore, a new tokenization scheme for Sindhi text is implemented (Mahar, *et al*., 2010).

3.2 Probability Calculation Process

The system takes the unigram, bigram, trigram and diacritic bigram probabilities of each diacritized word from probability tables and then performs multiplication operation on obtained probability values and return diacritic word having highest probability value. If word bigram or word trigram probability value is zero then due to the multiplication operation, the whole result will be zero and system rejects this particular diacritized word.

Consider the poem from our proposed corpus:

سورَن سانڊياس، پورَن پالي آهيان،
سُکَن جي، سَيَدُ چَئي، پُکي نه پياس،
جَيڪُس آنئُن هُياس، گَري گونَدَرَ وَلَ جي.

Few words are found in this poem having ambiguity like سکن، جي، سيد، پکي، چئي، گري، گوندر, these words can be used in different meanings and sounds by changing diacritical marks. (Table 2) shows many homographic structures of these ambiguous words producing various sounds and meanings. Now if we want to restore the diacritic symbols of the second line of above poem:

سکن جي سيد چئي پکي نه پياس

The system selects the first word from right to left that is نڪس and compares the pattern of this non-diacritic word with the corpus of diacritic words. System found two types of words سُکَن and سِکَن and takes unigram probabilities of these words from p_x, bigram probabilities from p_y, trigram probabilities from p_z and diacritic bigram probability value from p_w and then performs multiplication operation on obtained UG, WB, WT and DB probabilities:

Word	UG	WB	WT	DB	Multiplication
سُکَن	0.00029	0.25	0.125	(0.21008+0.25233)	**4.19 x 10^{-06}**
سِکَن	0.00003	0.00	0.0	(0.17617+0.25233)	0.00

As word سُکَن has high probability value than word سِکَن therefore system returns this diacritic word at the same location of selected word.

Then system selects second word that is يج then compares the structure of selected non-diacritic word with diacritic words and found four types of words جيَ، جيُ، جيِ، جَيْ and then takes unigram probabilities of these words from p_x, bigram probabilities from p_y, trigram probabilities of these words from p_z and diacritic bigram probability value from p_w then multiplies the obtained probabilities, calculated results are given below. System returns diacritic word جيْ at the same location of selected non-diacritic word because it has high probability than others.

Word	UG	WB	WT	DB	Multiplication
جيَ	0.00021	0.00	0.0	0.17617	0.00
جيُ	0.00043	0.00	0.0	0.39896	0.00
جَيْ	0.00047	0.1538	0.0769	0.18925	1.05 x 10^{-06}
جيْ	0.00058	0.3333	0.25	0.18652	**9.01 x 10^{-06}**

The system selects third word ديس from the line and finds two types of word سَيَدُ ، دَيَس and takes into the same process, after passing through which multiplies UG, WB, WT and DB probabilities of each word:

Word	UG	WB	WT	DB	Multiplication
سَيَدُ	0.00138	0.9473	0.0263	(0.27336+0.21729)	**1.69 x 10^{-05}**
سَيَدَ	0.00014	0.00	0.0	(0.27336+0.27336)	0.00

System returns word سَيَدُ because it has high probability value than word سَيَدَ .
Respectively selecting fourth word يئچ from selected line and finds two types of word چَئي ، چَئِي. The process repeats and finally following results are generated:

Word	UG	WB	WT	DB	Multiplication
چَئي	0.00588	0.0062	0.0062	0.27336	**6.18 x 10^{-08}**
چَئِي	0.00021	0.00	0.0	0.25233	0.00

System returns word چَئي at the same place of non-diacritic because it has high probability value than word چَئِي .

Consecutively fifth word پـکـي is selected from the line, system finds three types of word پِکِي ، پَکِي ، پُکِي. Passes through the procedure of UG, WB, WT and DB multiplications, the results are as under:

Word	UG	WB	WT	DB	Multiplication
پُکِي	0.00003	1.0	1.0	0.39896	**1.19 x 10^{-05}**
پَکِي	0.00025	0.00	0.0	0.17617	0.00
پِکِي	0.0007	0.00	0.0	0.23834	0.00

System returns word پُکِي because it has high probability value.

Now system selects sixth and seventh words من, سايپ and found only one type of each word therefore system considers this word having no ambiguity and takes unigram probability value of each word returns their equivalent diacritic words.

3.3 Viterbi Algorithm

The Viterbi algorithm is used to find the most likely path transitions, when more than one homographic word occur then the path of diacritized words having largest probability value (multiplication of calculated n-grams) is selected as the most likely path.

The probability of sequence of words is stored in an array called DATA (i, k), where i is the total number of words and k is the number of homographic words, the maximum value of k has been found 10 in this corpus.

Consider the sentence نـکـس جـي سـيـ چـئـي پـکـي من سايپ, in order to the implementation of the viterbi algorithm. The algorithm selects one diacritized word having highest probability value from the given list of homographic words. The calculated probabilities of homographic words نـکـس, جـي and others are processed as follows:

DATA (i, k) = P(ZB | PS) x P(سُکَنَ) x P(جـي | سُکَنَ) x P(سـي | جـي | سُکَنَ), where k = 1, 2, …, 10.
DATA (i, k) = P(ZB | ZR) x P(سِکَنَ) x P(جـي | سِکَنَ) x P(سـي | جـي | سِکَنَ)

So that DATA (1, 1) = 4.19 x 10^{-06} and DATA (1, 2) = 0.00

In the second part of the algorithm, both calculated values are compared and selected largest one and then store in the first location of PTR.

In the second iteration of algorithm where i=2 and initially k=1

DATA (i, k) = P(ZB | ZR) x P(جِيَ) x P(سـي | جِيَ) x P(چـئـي | سـي | جِيَ)
DATA (i, k) = P(PS | ZR) x P(جِيُ) x P(سـي | جِيُ) x P(چـئـي | سـي | جِيُ)
DATA (i, k) = P(JZ | ZB) x P(جَيْ) x P(سـي | جَيْ) x P(چـئـي | سـي | جَيْ)
DATA (i, k) = P(JZ | ZR) x P(جِيْ) x P(سـي | جِيْ) x P(چـئـي | سـي | جِيْ)

So that DATA (2, 1) = 0.00, DATA (2, 2) = 0.00, DATA (2, 3) = 1.05 x 10^{-06} and DATA (2, 4) = 9.01 x 10^{-06}

All calculated values are compared in the second part of the algorithm and system selects the largest one and store into the second location of PTR.

This process continues up to the last word of the given text. Finally system display all stored probability values from 1 to N with their equivalent associated words. The graphical model of path transition is depicted in **(Fig. 1)**.

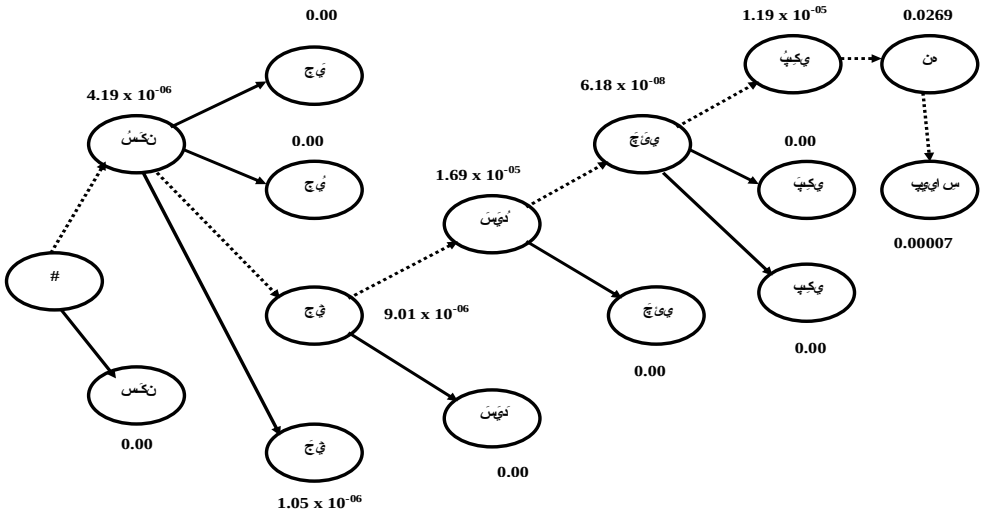

Fig.1. The model shows an example of a path transition through viterbi algorithm.

VB.NET frame work is used for the back end and front end development of diacritic restoration system. As Sindhi script is written from right to left; the cursor is flashed from right to left direction therefore for Sindhi writing system, we install MB Sindhi software (Majid, 2006), for the correct representation and visualization MB Lateefi is selected as default font style.

RESULTS AND DISCUSSIONS

4.1 Probability Tables

Two types of words are available in our proposed corpora: (a) words having no ambiguity like جا، ۽، مان (b) words having ambiguity, ambiguous are the words having multiple homographic structures due to placement of different diacritic symbols like کن، قسم، پٽ . For each word, system compares the structure of non-diacritized word with diacritized words and then computes the probabilities of each diacritized word. For words having no ambiguity, system calculates the frequency of selected word by using unigram model then recognizes the pattern of non-diacritized input word with diacritized word and then fixes the diacritized word at the same location.

For those words having ambiguity, the system automatically generates probability table p_x of all words by using unigram model, **(Table 2)** shows the sample of calculated probabilities. System compares the structure of non-diacritized word with diacritized words and selects diacritized word having high probability value, the diacritized word accuracy of 81.4% was achieved by using this model but selection of word having higher probability is not always correct, for instance, consider the line سر ڊونڊيا، ڏڙ نه لهان، ڏڙ ڊونڊيا، سر نَاه from L^{test}, system selects word سرَ instead of word سرُ because word سرَ has high probability value but the correct word is سرُ.

Therefore, for getting more accuracy, system computes the probabilities of words bigram and automatically generates probability table p_y by using eq.2, the sample of calculated words bigram probabilities are shown in **(Table 3)**. We have not achieved required results with words bigram, therefore, system also computes the probabilities of words trigram and generates probability table p_z by using eq. 3, **(Table 4)**. The system also computes the probabilities of diacritic sequence using bigram model (see eq. 4) and generates probability table's f_w and p_w, **(Table 5)** shows the sample of calculated frequencies and **(Table 6)** shows the sample of calculated probabilities. The proposed mechanism is mainly based on the multiplications of Diacritic Bigram (DB), Unigram (UG), Words Bigram (WB) and Words Trigram (WT) probabilities (see eq. 5).

Table 2: The sample of words frequencies with their equivalent probabilities

Word	F	P	Word	F	P	Word	F	P
سُکَن	8	0.00029	پَکي	7	0.00025	پالي	2	0.00007
سِکَن	1	0.00003	پِکي	2	0.00007	اَهيان	36	0.00131
جيَ	6	0.00021	ڇَني	161	0.00588	جَيڪُس	10	0.00036
جيُ	12	0.00043	ڇَنِي	6	0.00021	اَڻون	101	0.00369
جَيْ	13	0.00047	پِياس	2	0.00007	هُياس	4	0.00014
جِيْ	16	0.00058	نه	736	0.0269	گَري	1	0.00003
سَيَدُ	38	0.00138	سوُرَن	21	0.00076	گِري	1	0.00003
سَيَدَ	4	0.00014	سانڊِپياس	1	0.00003	گوندرَ	16	0.00058
پُکِي	1	0.00003	پوُرَن	2	0.00007	گوندرُ	1	0.00003

Table 3: The sample of words bigram frequencies and their equivalent probabilities

Word Bigram	F	P	Word Bigram	f	P
سوُرَن سانڊِپياس	1	0.048	ڇَني پُکِي	0	0.00
سانڊِپياس پوُرَن	1	1.00	پُکِي نه	1	1.0
پوُرَن پالي	1	0.5	پَکِي نه	0	0.00
پالي اَهيان	1	0.5	پِکي نه	0	0.00
سُکَن جيْ	2	0.25	نه پِياس	1	0.0013
سِکَن جيْ	0	0.00	جَيڪُس اَڻون	1	0.1
جيْ سَيَدُ	4	0.3333	اَڻون هُياس	1	0.0099
جَي سَيَدُ	2	0.1538	هُياس گَري	1	0.25
جيَ سَيَدُ	0	0.00	گَري گوُندرَ	1	1.0
جيِ سَيَدُ	0	0.00	گَرِي گوُندرَ	0	0.00
سَيَدُ ڇَني	36	0.9473	گوندرَ وَل	1	0.0625
سَيَدَ ڇَني	0	0.00	گوُندرُ وَل	0	0.00
ڇَني پُکِي	1	0.0062	وَل جَيْ	1	0.3333

Table 4: The sample of word trigram frequencies and their equivalent probabilities

Word trigram	F	P	Word trigram	f	P
سوُرَن سانڌِيياس پوُرَن	1	0.0476	چَئِي پُڪِي نه	1	0.0062
سانڌِيياس پوُرَن پالِي	1	1.0	چَئِي پُڪِي نه	0	0.0
پوُرَن پالِي آهِيان	1	0.5	پُڪِي نه پِيياس	1	1.0
سُڳَن جيۡ سَيَدُ	1	0.125	پَڪِي نه پِيياس	0	0.0
سِڳَن جيۡ سَيَدُ	0	0.0	پِڪِي نه پِيياس	0	0.0
جيۡ سَيَدُ چَئِي	4	0.25	جَيڪُس آنوُن هُياس	1	0.1
جَي سَيَدُ چَئِي	1	0.0769	آنوُن هُياس گَري	1	0.0099
جيَ سَيَدُ چَئِي	0	0.0	گَري گوُندَرَ وَلِي	1	1.0
جيُ سَيَدُ چَئِي	0	0.0	گَري گوُندَرَ وَل	0	0.0
سَيَدُ چَئِي پُڪِي	1	0.0263	گوُندَرَ وَل جِي	1	0.0625
سَيَدَ چَئِي پُڪِي	0	0.0	گوُندَرُ وَل جِي	0	0.0

Table 5: Bigram diacritic frequencies from the proposed corpus.

i-1\i	#	ZB	ZR	PS	JZ	BZ	SD	Sum
#	0	163	73	51	0	0	0	287
ZB	0	117	108	93	81	4	25	428
ZR	0	34	46	77	36	0	0	193
PS	0	25	54	12	28	0	0	119
JZ	0	21	7	5	0	0	0	33
BZ	0	0	0	0	0	0	0	0
SD	0	8	2	6	4	0	0	20

Table 6: Biagram diacritic probabilities from proposed corpus

i-1\i	#	ZB	ZR	PS	JZ	BZ	SD
#	0	0.56794	0.25436	0.17770	0.0	0.0	0.0
ZB	0	0.27336	0.25233	0.21729	0.18925	0.00934	0.058411
ZR	0	0.17617	0.23834	0.39896	0.18652	0.0	0.0
PS	0	0.21008	0.45378	0.10084	0.23529	0.0	0.0
JZ	0	0.63636	0.21212	0.15151	0.0	0.0	0.0
BZ	0	0.0	0.0	0.0	0.0	0.0	0.0
SD	0	0.4	0.1	0.3	0.2	0.0	0.0

4.2 Performance Measurements

For training and testing, the corpus L is classifying into two parts: (i) $L^{training}$ that contain 27360 words and (ii) L^{test} that contains 3919 words. The testing corpus is selected from training corpus which is approximately 14.32% of training data. The diacritic restoration system was tested using randomly selected 250 poems from $L^{training}$. The system is evaluated to measure the accuracy of words in terms of percentage in the L^{test}. For all experiments, two types of error rates are reported in order to calculate the performance of system, one is word error rate (WER) for the percentage of incorrectly delimited words due to hard white space, the other one is diacritization error rate (DER) for proportion of incorrectly restored diacritics, the calculated results are shown in **(Table 7)**.

Table 7: Calculated results of two word types

Types of Words	No. of Words	No. of Characters	WER%	DER %
Words having no ambiguity	2936	12137	0.53	0.27
Words having ambiguity	983	4066	0.18	2.94
Total	3919	16203	0.71	3.21

4.3 Discussions

Various approaches have been proposed and applied for diacritic restorations of Arabic script based written languages like Hidden Markov Model (HMM), Maximum Entropy (ME), Neural Network (NN), Finite State Transducers (FST), combination of different Linguistic Features (LF), Word Net (WN) however, for any diacritic based language, no studies have explored the optimal level of diacritization yet. Most of the researchers (Gal, 2002; Harby, *et al*., 2008; Elshafei, *et al.*, 2006) used HMM approach for diacritic restoration and reported 81% to 95.2% accuracy of system's performance. During literature survey, it has been found that the diacritization error rate is reported by different researchers by using different approaches from 4.1% to 6.35% and the word error rate is reported from 5.5% to 17.3%, the DER and WER of different researchers are shown in **(Table 8)**.

Table 8: DER and WER reported in literature.

Proposed Approach	DER%	WER%
HMM (Elshafei, 2006)	4.1	5.5
ME (Zitouni, 2006)	5.1	17.3
FST (Nelken, 2005)	6.35	7.33
LF (Habash, 2007)	4.8	14.9
WN (Mahar, 2011)	3.39	0.71
Proposed Method	3.21	0.71

The DER about 3.21% is achieved by using proposed mechanism, this error rate is lower than any error rate reported previously in the literature (the claim of error rate is based on our developed corpora, this may not true if a different corpus of any other language is used). **(Fig. 2)** shows the graphical representation of WER and DER of various proposed approaches for diacritic restorations. The diacritic error rate of words having no ambiguity is 0.27% due to presence of embedded space in the compound words; if words are managed properly then DER could be further reduced about 2.94%. Similarly, the system could probably be improved using Laplace or Katz back-off smoothing to eliminate the multiplications by 0 that arise when using a small training corpus or higher-order n-grams.

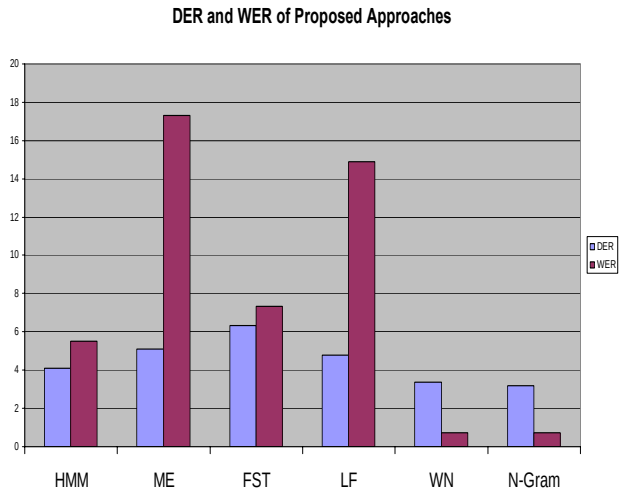

Fig. 2. Graphical evaluation of WER and DER of various proposed approaches

Approximately, 389 Arabic words and few Persian and Urdu words are present in Shah Jo Risalo, for instance, consider one line from the proposed corpus لَا يُكَلِّفُ اللّٰهُ نَفسًا اِلاّ وُسعَهَا، جيڪا پُڄندِيَم سا from right to left six words are selected from Quran (a holy book). Many words like عاشق، پتنگ، شهيد، شاه، ثواب are same in Urdu and Sindhi; we have also achieved acceptable level results with words of these languages therefore proposed methodology can be used for Arabic, Urdu and Persian.

The proposed system is focused on six diacritic symbols, among them three symbols are short vowels that represented by letters "ə", "i", "u" which are corresponding to "Zabara", "Zair", "Pesho" respectively and other three are represented by symbols " ّ", " ٖ ", " ٗ" which are corresponding to "Shadda", "Baa Zabara", "Jazam" respectively.

Two mandatory steps are always required for diacritics restoration system of any language: (1) the identification of missing diacritics before filling in a given word and (2) how diacritics are restored and on what basis. Through our proposed mechanism, identification of missing diacritics is not necessary because each non-diacritic word is replaced with diacritic word and diacritic word is restored on the basis of high probability value.

CONCLUSION

The N-gram based diacritic restoration system using Viterbi algorithm was proposed for Sindhi language. The orthographical system of Arabic script based written languages is highly homographic and the text is written without diacritic symbols, this creates morphological and lexical ambiguity and it is difficult to pronounce non-diacritized text correctly. In this paper, automatic system of diacritics restoration for Sindhi text using different N-gram models i.e., unigram, bigram and trigram is discussed and represented. The acceptable results are achieved with these three kinds of N-gram models but 4-gram or also 5-gram may be useful for different kinds of Sindhi corpora. The corpus of the language is necessary for diacritic restoration system therefore Shah Jo Risalo is used as corpora. The word error rate of 0.71% and diacritic error rate of 3.21% have been achieved. The possible future work is to train large corpora using higher-order n-grams, the testing data will select from different corpus, and smoothing method will used to avoid 0 probabilities of n-grams.

REFERENCE

Aadvani., K. (2009) Shah Jo Risalo,2nd Edition, Sindhica Academy, Karachi, Pakistan.

Elshafei, M., H. A. Muhtaseb and M. Alghamdi (2006) Statistical Methods for Automatic Diacritization of Arabic text, Proceedings 18th National Computer Conference, Riyadh.18:301-306.

Gal, Y., (2002) An HMM Approach to Vowel restoration in Arabic and Hebrew. Proceedings of the ACL-02 Workshop on Computational Approaches to Semitic Languages, Association for Computational Linguistic, Philadelphia, Pennsylvania, 1-7.

Habash, N. and O. Rambow (2007) Arabic Diacritization through Full Morphological Tagging, Proceedings of the North American Chapter of the Association for Computational Linguistic, Association for Computational Linguistic, Rochester, New York, 53-56.

Harby, A. A., M. A. Shehawey and R. S. Barogy (2008) A Statistical Approach for Quran Vowel Restoration, ICGST-AIML 8 (**3**): 9-16.

Jurafsky, D., J. H. Martin (2000) Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistic and Speech Recognition, Prentice-Hall, New Jersey, 176-206.

Mahar, J. A. and G. Q. Memon (2009) Phonology for Sindhi Letter to Sound Conversion, Journal of Information and Communication Technology 3 (**1**): 11-20.

Mahar, J. A. and G. Q. Memon (2010) Sindhi Part of Speech Tagging System using WordNet, International Journal of Computer Theory and Eng. 2 (**4**): 538-545.

Mahar, J. A. and G. Q. Memon (2011) Lexicon Based Diacritic Restorations using WordNet for Sindhi, International Jou. of Academic Research, 3 (**2**):37-43.

Majid, B., (2006) www.fileguru.com/apps/mb-sindhi-software.

Nelken, R. and S. M. Shieber (2005) Arabic Diacritization Using Weighted Finite-State Transducers, Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, Association for Computational Linguistic, Ann Arbor, Michigan, 79-86.

Rashwan M, M. Albadrashiny, M. Attia, S. Abdou and A. Rafea (2011) A Stochastic Arabic Diacritizer Based on a Hybrid of Factorized and Unfactorized Textual Features, IEEE Transactions on Audio, Speech and Language Processing, Vol.19. Number (**1**): 166-175.

Roth, R., O. Rambow and N. Habash (2008) Arabic Morphological Tagging, Diacritization and Lemmatization Using Lexeme Models and Feature Ranking, Proceedings of ACL HLT, 117-120.

Sultan, H. (2001) Automatic Arabic Diacritization using Neural Network, Scientific Bulletin of Faculty of Engineering Ain-Shams University: Electrical Engineering, 36 (**4**): 501-510.

Zitouni, I., J. S. Sorensen and R. Sarikaya (2006) Maximum Entropy Based Restoration of Arabic Diacritics, Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, 577-584.