# Design and Analysis of an Effective Corpus for Evaluation of Bengali Text Compression Schemes

Md. Rafiqul Islam
Computer Science and Engineering Discipline, Khulna University, Khulna, Bangladesh.
Email: dmri1978@yahoo.com

S. A. Ahsan Rajon
Computer Science and Engineering Discipline, Khulna University, Khulna, Bangladesh.
Email: ahsan.rajon@gmail.com

*Abstract* — **In this paper, we propose an effective platform for evaluation of Bengali text compression schemes. A novel scheme for construction of Bengali text compression corpus has also been incorporated in this paper. A methodical study on the formulation-approaches of text corpus for data compression and present an effective corpus named Ekushe-Khul for evaluating the Bengali text compression schemes has also been presented in this paper. To design the Bengali text compression corpus, Type to Token Ratio has been considered as the selection criteria with a number of secondary considerations. This paper also presents a mathematical analysis on data compression performance with structural aspects of corpora. A comprehensive analysis on the evolving criteria of text compression corpora with related issues in designing dictionary based compression are extensively incorporated here. The proposed corpus is effective for evaluating compression efficiency of small and middle sized Bengali text files.**

*Index Terms* — **Corpus, Bengali Text, Bengali Text Compression, Dictionary Coding, Data Management, Evaluation Platform, Compression Efficiency, Type to Token Ratio (TTR).**

## I. Introduction

With a number of languages in the world, Bengali is the only language, which has been established at a cost of lives. However, in the establishment of Bengali as a glittering candidate in the field of research the number of steps or achievements is not so mention-worthy. Enhancing Data Management techniques for Bengali text has not yet got any robust base. Text Compression, which is one of the important aspects of elementary data management and text manipulation schemes, is still now mostly based on generalized universal compression techniques. The remaining sophisticated Bengali text compression schemes also suffer from unavailability of standard evaluation platform, *i.e.* unavailability of text compression corpus. This paper proposes a text compression corpus for evaluating Bengali text compression schemes.

The construction of a data compression corpus is now a demand-of-time. The state-of-the-art technologies of data management have already equipped the benchmarks and standard collection of experimental data sets for performance evaluation [1]. In Bengali, though a highly impressive step has been initiated in [2] for constructing a linguistic corpus having CIIL (Central Institute of Indian Languages) corpus [4] as a pioneer, still now any mentionable sophisticated data compression corpus is not available with text compression benchmarks. This paper focuses on designing a Bengali text compression corpus based on *Type to Token Ratio* (*TTR*) and provides the mathematical framework for various aspects of choosing the corpus with compression benchmarks.

The paper is targeted to design a corpus that will facilitate the future researchers especially in the research of Bengali text compression to have a test-bed for evaluating their performance. For designing such a corpus we are to derive a novel scheme of text compression corpus formation because of having no existing benchmark.

The paper is organized into eight sections. Section II provides elementary concepts on corpus and text compression schemes. The key-points for constructing a corpus are presented in section III. Section IV is devoted with the literature review presenting various aspects of the existing corpora (corpuses). Section V describes the proposed approach for formulating the corpus. A brief description on the files comprising the proposed text compression corpus is presented in section VI. Analysis of the proposed corpus is depicted in section VII. Various

This paper is an enhancement of the paper "On the Design of an Effective Corpus for Evaluation of Bengali Text Compression Schemes", by Md. Rafiqul Islam, and S. A. Ahsan Rajon, which was published in the proceedings of International Conference on Computer and Information Technology (ICCIT' 2008), 25-27 December 2008, Khulna, Bangladesh.

Contact Author: S. A. Ahsan Rajon

doi:10.4304/jcp.5.1.59-68

issues on the usability of the proposed corpus for other languages is presented in section VIII. The paper ends with section IX providing conclusion and recommendation.

## II. Overview OF Corpus and Data Compression

Text compression is an elementary aspect of data compression. Compression is the process of reducing the size of a file or data by identifying and removing redundancy in its structure [7]. Data Compression offers an effective approach of reducing communication costs by using available bandwidth effectively. Data Compression technique is generally divided into two categories; namely, Lossless Data Compression and Lossy Data Compression. For lossless schemes, the recovery of data should be exact. Lossless compression algorithms are to some extent essential for all kinds of text processing, scientific and statistical databases applications, medical and biological image processing, DNA and other biological data management and so on. However, a lossy data compression technique does not ensure the exact recovery of data. For image compression and multimedia data compression, there is a great use of lossy data compression. From the early 1990, certain specific platforms of evaluating English text compression schemes are adapted for standard researches, which are better known as data compression corpus [1].

In general sense, a corpus is simply a collection of texts. From the point of view of data compression, compression-corpus is a standard collection of texts or data for analyzing and evaluating effectiveness and efficiency *i.e.* performance of compression schemes. The mostly used data compression corpora for evaluating data compression are namely Calgary corpus, Canterbury corpus and Project Gutenberg. Though the corpora for English were introduced in the early 1990, still now, for analyzing and evaluating Bengali text compression performance, no such complete and standard corpus is available. In this paper, we present a novel approach for designing data compression corpus for evaluation of Bengali text compression scheme. Implementation of the proposed scheme has also been focused in the developed *Ekushe-Khul* corpus.

For analyzing domain specific or language specific text-compression scheme, it is a must to have the domain oriented or language oriented benchmark to analyze the compression schemes. There are a number of English text compression corpora and corresponding benchmarks which are extensively used for evaluation of English Text compression schemes. There are also a number of multilingual text corpora involving English for evaluating English text compression with respect to other languages. But in case of Bengali there is neither any corpus for evaluation of text compression performance nor any benchmark for the same. A detail explanation on the necessity of Language specific corpora and the suitability (or unsuitability) of any corpora for evaluating non-English concerns is provided by Sarker and Roeck in [10] with a comprehensive explanation by N. S. Dash in [12] and by Akshar et al. in [11]. The required changes for designing a new corpus for Bengali in order to meet new demands of computational linguistics and other approaches is elaborately provided in [9].

## III. Characteristics of a Data Compression Corpus

The characteristics of a corpus are defined by the purpose of creating the corpus and the evaluation arena for which the corpus is designed for. These criteria give rise to the necessity to field-specific corpus like data compression corpus, information retrieval corpus, corpus for data mining research etc. According to Arnold *et al.*, there are six criteria for choosing a corpus [1]. Firstly, the files should be chosen as representative of the files used and expected to be used in compression method. Secondly, the availability of the file should be ensured. Availability of public domain materials is the third consideration for the corpus. They also suggest that the files should not be larger than necessary and it should be *perceived* to be valid and useful. Finally, the corpus should be *actually* valid and useful.

The mentioned criteria for choosing corpus have successfully established *Canterbury Corpus* as one of the extensively adapted corpus for data compression. However, due to the rapid growth of technology, additional criteria have been evolved in constructing the corpus. Firstly, the criteria were specified for evaluating the compression of middle-sized files. But, with the advancement of embedded devices with lower processing speed and small memory (like mobile-phones) it has become extremely important to provide a new test-bed for evaluation of small text compression schemes. Frequent uses of mobile phones, short text messages and emails have already made a transformation on the use of texts, and hence necessity of field and size specific corpus formation has been evolved. Taking these limitations into concern, we provide specific files regarding the issues with consideration of other traditional characteristics. Moreover, the file structure *i.e.* sentence structure was not considered as any important criteria for choosing corpora. But, compression ratio gradually varies with variations of sentence construction and file structure. Besides these, probability of occurrences of texts-components and frequency of those are important criteria, which fluctuate the text compression performance.

## IV. Literature Review

The history of mostly adapted data compression corpus started a long ago. *Calgary Corpus*, which was collected in 1987 and was published in 1990 is considered as the pioneer of data compression corpus. The files in *Calgary corpus* [1] with their content category are listed in table I.

TABLE I

FILES IN THE *CALGARY CORPUS*

| Title | Description |
|---|---|
| bib | Bibliographic File |
| book1 | Hardy: Far From the Madding crowd |
| book2 | Witten: Principles of computer speech |
| geo | Geophysical data |
| news | News batch file |
| obj1 | Compiled Code for Vax |
| obj2 | Compiled Code for Apple Macintosh |
| paper1 | Witten: Arithmetic Coding for compression |
| paper2 | Witten: Computer Security |
| paper3 | Witten: In Search of Autonomy |
| paper4 | Cleary: Programming by example revisited |
| paper5 | Cleary: logical implementation of arithmetic |
| paper6 | Cleary: Compact Hash tables |
| pic | Picture from CCITT Facsimile |
| progc | C source code |
| progl | Lisp source code |
| progp | Pascal Source Code |
| trans | Transcript of a session on terminal |

The next initiative and the mostly adapted one is *Canterbury Corpus*. It was developed by Arnold *et al.* [1]. Canterbury corpus considered the benchmark derived from the Calgary corpus as their corpus development basement. In Table II, the files constituting Canterbury corpus are presented.

TABLE II

FILES IN THE *CANTERBURY CORPUS*

| Title | Description |
|---|---|
| alice29.txt | English text |
| ptt5 | Fax images |
| fields.c | C source code |
| kennedy.xls | Spread-sheet Files |
| Ssum | SPARC Executable |
| lcet10.txt | Technical documents |
| plrabn12.txt | English Poetry |
| cp.html | HTML |
| grammar.lsp | Lisp source code |
| xargs.1 | GNU Manual pages |
| asyoulik.txt | plays |

The above files are solely for evaluation of English text compression evaluation corpus and hence not effective or at all applicable for evaluating Bengali text compression.

The most recent and extremely organized analysis on Bengali news corpus is provided by Majumder *et al.* [2]. This paper proposes a new corpus on the basis of the well circulated newspaper *prothom-alo* [5] and provides exhaustive analysis regarding *TTR* (Type-to-Token-Ratio), Function Word and other morphological and linguistic analysis. For analysis of data compression efficiency, this corpus requires some specific modification in the sense that, data compression involves the additional criteria of heterogeneity and text-size. Moreover, the news corpus proposed in [2] is extensively tested for linguistic analysis only. In order to use this for data compression, further text-compression related analysis is a must. According to their methodology, they collect html news files from *prothom-alo* websites and converting it to Unicode, they categorize the total collection into twenty seven distinct groups, which are also available in a single file of size three hundred and eighteen mega bytes. For most practical applications and researches on middle-sized or small text compression, it is rarely necessary to have larger files, rather it is essential to have standard small and medium sized corpus.

The fist corpus in Bengali developed by Central Institute of Indian Languages (CIIL) in the years 1991-1995 possesses a collection of three million Bengali words, which provide valuable linguistic data for research on Bengali language analysis [4]. In evaluating compression efficiency, this impressive corpus also suffers from the limitation of heterogeneity and lack of embedded and small device supportability.

Besides the corpora presented there are a number of researches available describing the features of Bengali linguistic corpus. A number of works is also devoted to have corpora on Bengali OCR (Optical Character Recognition). The contribution of B. B. Chaudhury is pioneer in this specific field of Bengali OCR formation [13].

The necessity of devising specific and relevant criteria for building Bengali corpora and the basic limitations as well as pitfalls of existing non-Bengali corpora building concepts are extensively incorporated by N. S. Dash in [9]. Prime techniques of Bengali text corpus processing, which are used for various linguistic activities including far more complicated orthographical, morphological, and lexicological concerns is also presented in [9]. Various criteria including Concordance of Words, Lexical Collocation, Key-Word-In-Context, Local Word Grouping, Lemmatization of Words, Parsing Sentences etc are widely described by N. S. Dash in [9]. Being the overall aspect of the literature in [9] is to furnish linguistic and information retrieval corpora, there is little usability of the technique [9] in text compression corpora formation.

The essentiality of domain specific corpora has motivated us to design Bengali text compression evaluation corpus. The unavailability of reference or benchmark in the field has also leaded us to develop a new concept for designing a corpus.

## V. Proposed Corpus for Evaluation of Bengali Text Compression Schemes

We propose a new corpus for evaluation of Bengali text compression schemes namely *Ekushe-Khul* corpus [8]. This corpus is intended for evaluating only short and middle-sized text compression schemes rather than evaluation for large text compression. One of the worth-mentionable points is, this corpus is selected by applying a new approach, which takes both *TTR* and *Compression Ratio (CR)* into account. The two concepts have been adapted because we do not have any standard and sophisticated Bengali text compression corpus available with which we may incorporate and compare the performance and thus perform the selection from candidate files as done for other existing data compression corpora.

To form the corpus we have considered ten groups- Articles, Poems, Advertisements, Speeches, News, Sms, Emails, Particulars, Stories And Reports. The files are of sized 4 KB to 1800 KB. The steps of constructing the proposed corpus are depicted in Fig 1.

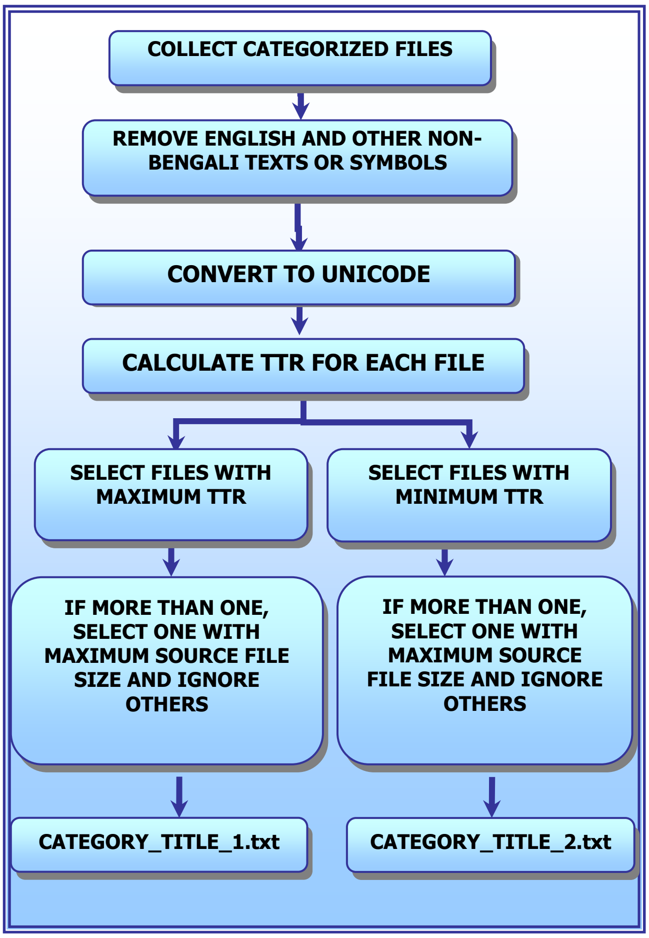
COLLECT CATEGORIZED FILES
REMOVE ENGLISH AND OTHER NON-BENGALI TEXTS OR SYMBOLS
CONVERT TO UNICODE
CALCULATE TTR FOR EACH FILE
SELECT FILES WITH MAXIMUM TTR
SELECT FILES WITH MINIMUM TTR
IF MORE THAN ONE, SELECT ONE WITH MAXIMUM SOURCE FILE SIZE AND IGNORE OTHERS
IF MORE THAN ONE, SELECT ONE WITH MAXIMUM SOURCE FILE SIZE AND IGNORE OTHERS
CATEGORY_TITLE_1.txt
CATEGORY_TITLE_2.txt

Fig. 1: Steps for constructing the Proposed Bengali text compression corpus.

The files constituting the proposed *Ekushe-Khul* corpus are shown in Table III. A detail description on the proposed corpus selection criteria with potential issues and concerned data compression parameters that we have proposed for developing Bengali text corpora for evaluation of Bengali text compression schemes is presented in section VI.

TABLE III
FILES CONSTITUTING THE PROPOSED *EKUSHEY-KHUL* CORPUS.

| File Name | Type |
|---|---|
| Article1.txt<br>Article2.txt | Articles in Bengali |
| Poem1.txt<br>Poem2.txt | Bengali Classic Poems |
| Advertise1.txt<br>Advertise2.txt | Advertisements |
| Speech1.txt<br>Speech2.txt | Various Speeches |
| News1.txt<br>News2.txt | News from Newspaper |
| SMS1.txt<br>SMS2.txt | Short Bengali Message |
| Email1.txt<br>Email2.txt | Emails in Bengali |
| Particulars1.txt<br>Particulars2.txt | Particulars |
| Story1.txt<br>Story2.txt | Bengali Stories |
| Report1.txt<br>Report2.txt | Academic Reports |

To collect the files, the first step was to remove redundancies and other components (*e.g.* html tags, letters and words from other languages, formatting *etc*.) and convert them into Unicode as several documents were non-standard encoded and saved as Unicode text files. These files are then forwarded for further analysis.

For each category of files, all the files are passed to a *TTR Calculator* and then *Compression Ratio Evaluator* for selection of final documents. In case of *Canterbury Corpus,* which is the pioneer for data compression corpus was formed by comparing each file with the standard of the *Calgary Corpus* of same group or category, and the file demonstrating the closest match in terms of compression performance was selected. But, as no such standard exists, we had to take account of both the criteria and hence selected two with the maximum and minimum consistence.

*TTR* is a measurement of how many times a previously encountered words repeat themselves before a new word makes its appearance in the text [2]. It is note worthy that though this measure is essential for language engineering, its application in data compression is not mandatory. For calculative approaches, *TTR* is calculated by dividing the total number of word tokens by the total number of distinct words. Texts with a large number of distinct words result into lower *TTR*. For this case, the task of forming a data compression dictionary appears to be more accurate. Conversely, for texts with greater *TTR i.e.* with greater number of frequent words (resulting lower number of distinct words) provide a suitable arena for constructing and evaluating dictionary-based and repetition-analysis oriented text compression schemes. This is the reason, which inspires us to provide peer-files for each category. The files with suffix '2' indicate the file with lower *TTR* (suitable for text compression fluctuation evaluation) and files with suffix '1' indicate greater *TTR* (suitable for dictionary-based text compression scheme evaluation) for the respective group.

The next step involves analyzing the files in terms of compression ratio [6] to establish logical and implementational validity of choosing *TTR* as a benchmark for constructing the corpus. Though a number of approaches exist for compression of English texts, (like Burrows-wheeler Transform [4], Arithmetic Coding, Huffman Coding etc.) it is a sad tale that, still now a little research has been dedicated for Bengali text compression schemes. The uses of non-sophisticated text compression schemes do not provide optimal performance for Bengali text compression purposes. For providing a benchmark of the proposed corpus, we adapt the dictionary based compression scheme in [7] and demonstrate the inter-relation of *TTR* with *Compression Ratio*.

Unavailability of existing sophisticated and multidimensional text Bengali text compression approach, we had a little alternatives to compare the performance fluctuation of our corpora. Though there are a number of English text compression schemes available, testing with those may never provide a comprehensive view to out developed scheme with implemented prototype.

## VI. Aspects of Corpus Selection Criteria and Text Compression Performance

Let *n* be the total number categories of collections of texts. The categories are representative of texts bearing distinct themes with differences in sentence construction, sentence wording and sentence structure.

Let the categories are $c_1, c_2, c_3, \ldots\ldots, c_n$ respectively.

For category $c_1$ the files are

$f_{1,1},\ f_{1,2},\ f_{1,3}, \ldots\ldots, f_{1,n}$

with length $l_{1,1},\ l_{1,2},\ l_{1,3}, \ldots\ldots, l_{1,n}$ , where, the length indicates total number of characters (including blank spaces, tabs, new lines and other punctuation marks).

If there are total of $w_{1,1}, w_{1,2}, w_{1,3}, \ldots\ldots, w_{1,n}$ words in $f_{1,1}, f_{1,2},\ f_{1,3}, \ldots\ldots, f_{1,n}$ respectively with number of distinct words be $d_{1,1},\ d_{1,2},\ d_{1,3}, \ldots\ldots, d_{1,n}$ for the same.

Here, for any file $f_{i,\,j}$, where *i* denotes *category index* and *j* denotes *file index, and* $w_{i\,j} \geq d_{i,j}$. Assume $T_{i.\,j}$ be the Type to Token Ratio (*TTR*) for file $f_{i,\,j}$.

That is, $T_{i,j} = \dfrac{w_{i,j}}{d_{i,j}}$.

For any two files *p, q* in category *i* , with *TTR* $T_{i,p}$ and $T_{1,q}$, if $T_{i,\,p} \geq T_{i,\,q}$ indicates that, $T_{i,\,p}$ contains larger number of distinct words than that of $T_{i,\,q}$.

Applying this rule, *TTRs* are counted for the files. Then the file with maximum and minimum *TTR*s are selected. For the file with maximum *TTR*, if the elements comprise the set $S_d$ and $S_t$ for distinct words and total words respectively, then for ideal case, set $S_d - S_t$ would tend towards *Φ* (*empty* or *null*) set.

Though *TTR* has been considered as a means of forming linguistic corpora and Natural Language Processing corpora including Information Retrieval corpora, taking it into concerns, it is possible to design Data Compression corpora too. Here, we provide an analysis has been presented on the effectiveness of using *TTR* as a criteria of constructing Corpora.

The basic relation between TTR and word-distribution is:

- If *TTR* is larger (greater) then, the number (frequency) of matching word is also greater; consequently, number of distinct words is smaller.
- If *TTR* is smaller (lower) then, the number (frequency) of matching word is smaller; consequently, number of distinct word is larger.

For designing dictionary based data compression approaches, *TTR* plays a great role. For constructing the dictionary, the prime concern is to analyze the probability of occurrence of each character or syllable or substring or words (or any combination of the mentioned units). The frequencies in which they occur are an important aspect of figuring the probability. As the total number of distinct words is greater in files with lower *TTR* value, the detection of the probability of syllables may be easier as the word distribution is approximately flat in those files. For building word-based dictionaries for data compression, files with higher *TTR* values are more effective as the recurring words may be very easily pointed out. The fluctuation may be easily identified by comparing the performance of the compression scheme using the peer files.

Let us consider a tertiary dictionary based text compression approach *DCA*, which uses a dictionary of *n* items. The total items consist of *c* characters, *s* syllables and *w* substrings, which save (in average) *sc*, *ss*, and *sw* bits each. Traditionally, $sw \geq\ ss \geq sc$ as words or substrings are generally considered as a collection of syllable and/or symbols and syllables are considered as a collection of characters, but the indices or pointers in most designs require consecutive values spanning against a threshold value. For a file with greater *TTR ( i.e.* Greater number of frequent words and lower number of distinct words*)*, the probability of encoding using word-based coding is high. Consequently, the bit-savings will also be high as the saving weightage is greater for words in compassion with other units. Let the replacement of number of distinct words be *rw*. Here, *rw* theoretically tends towards the total occurrence in average.

Therefore, the total savings in this phase is

$$S_1 = \sum_{y=1}^{rw} (sw_y \times fw_y)$$

where $fw$ is the frequency of corresponding distinct word. That is, $fw$ indicates the frequency of occurrence of distinct word-indexed at $y$ and ideally, $fw \rightarrow tw$.

The number of distinct syllable *rs* in a collection of highly distinct words should be lower. If the weightage is considered as double as distinct words, the total savings would be-

$$S_2 = \sum_{y=1}^{rs} (ss_y \times fs_y)$$

where *fs* is the frequency of the syllable.

The rest of the characters should be encoded separately resulting savings of

$$S_3 = \sum_{y=1}^{rc} (sc_y \times fc_y)$$

If we encode a total of *p* characters separately, which saves *sp* bits for each character and if we encode *q* words separately with an average of *d* characters per word, whenever even *d* = *p*, the savings will be much more greater for word based bit-savings, since the indexing of the constructed code base will be approximately sequential and hence, each single character and each single word based encoding will occupy about same length.

Thus, the total saving is,

$$S = S_1 + S_2 + S_3$$

$$S = \sum_{y=1}^{rw} (sw_y \times fw_y) + \sum_{y=1}^{rs} (ss_y \times fs_y) + \sum_{y=1}^{rc} (sc_y \times fc_y) \quad (1)$$

For a file with lower *TTR, (i.e.* greater number of distinct words and lower number of frequent words*)* there will have non-frequent occurrence of matching words and syllables. This will result extensive application and usage of character based encoding, as the primary unit of encoding is character. That is, $rw' < rw$ and $fw' \rightarrow 1$.

Similarly, $rs' < rs$ and Therefore, if there are total of $tw$ words in the file, then, $fs' \rightarrow 1$.

But, because of remaining large number of character units, $rc' \rightarrow tc'$.

If there a total of $tw'$ words in the file with lower *TTR* values, then,

$$S_1' = \sum_{y=1}^{rw'} (sw_y \times fw'_y)$$

and $fw' \rightarrow tw'$

As low frequent words would be encountered, as much frequent syllables would be faced. However, a minor portion of the syllables may be coded in form of words. That is, we will have a larger number of character units and syllable units ($ts'$ and $tc'$) left to be coded.

For syllable based coding,

$$S_2' = \sum_{y=1}^{rs'} (ss_y \times fs'_y)$$

After the two steps, a greater number of characters are left to be coded with distinct characters. Typically this value $tc'$ may be $tc' < ts'$ and for usual cases, $tc' < tw'$. Let these steps save,

$$S_3' = \sum_{y=1}^{rc'} (sc_y \times fc'_y)$$

$$S' = S'_1 + S'_2 + S'_3$$

$$S' = \sum_{y=1}^{rw'} (sw_y \times fw'_y) + \sum_{y=1}^{rs'} (ss_y \times fs'_y) + \sum_{y=1}^{rc'} (sc_y \times fc'_y) \quad (2)$$

It has been stated that, the ratio of bit-saving for word based , syllable based and character based encoding is 4:2:1 (*i.e.* for the ratios *a : b : c* we may write, *a > b > c*). The ratio is taken as standard because, for character based dictionary oriented encoding, single character or symbol is considered as the primary unit. Consequently, substrings and syllables ranges from two to any higher value. The number of characters that comprises any word may ideally be considered in the range of greater than four whereas short length words may be thought of as the subgroup of substrings. Hence, it can be easily remarked that, for the above consequences,

(i) $S_1 \geq S_1'$ ; since the number of word-matching is greater in $S_1$ and the difference is a multiplier of four.

(ii) $S_2 \geq S_2'$ ; since the number of syllable-matching is greater in $S_2$ and the difference is a multiplier of two.

(iii) $S_3 \leq S_3'$ ; since the number of matching-character is greater in $S_3$ and the difference is an unit multiplier.

From the above three consequences, we may easily point that, though $S_3' \geq S_3$ , it is an average case that, $S_3'$ will cross the saturation point or threshold value comprising of (*4 + 2 = 6*) times higher amplitude. Consequently, we may easily conclude: $S > S'$ .

It may be also proved the same as follows:

From the third clause we may write,

$$S_3 + \alpha = S_3' ,$$

where $\alpha$ is a constant bit-factor.

From clause *(i)* and *(ii)*, we can write

$$S_1 + S_2 \geq S_1' + S_2' .$$

From (i), (ii), we may deduce that (by adding the same value $S_3^{/}$ in both sides),

$$S_1 + S_2 + S_3' \geq S_1' + S_2' + S_3' \qquad (3)$$

Being a constant factor $\alpha = 1,2,3,........etc\,(number\ \ of\ \ bits)$, we may write (3) as,

$$S_1 + S_2 + S_3 + \alpha \geq S_1' + S_2' + S_3' \qquad (4)$$

It has been presented that,

$$S = S_1 + S_2 + S_3$$

and,

$$S' = S_1' + S_2' + S_3' .$$

Hence, from *(4)*,

$$S + \alpha \geq S' .$$

Recall from earlier explanation that, S and $S^{/}$ are the aggregation of bit savings for matching characters, syllables and words. Again, as the ratio of bit saving for word-based, syllable based and character based encoding is considered 4:2:1, consideration of linear transformation factor *l* indicates that, S will rush towards *6l* where $\alpha$ tends words *l*. for the average cases.

In better cases, as all the characters are likely to be pre-coded with either syllable based coding or word based coding section. After coding with word based and syllable based coding there will be little portion of source text left for character based coding. That is, the total saving is probable to be derived from word based saving.

Consequently, $S_1 + S_2 \rightarrow S$, whereas, character based saving will tend towards zero. $S_3 \rightarrow 0$ .

Again, being $S_3 + \alpha = S_3'$ ; $\alpha$ too will tend towards zero. That is, $\alpha \rightarrow 0$ .

Again, from clause (i), we get,

$$S_1 = S_1' + \alpha_1 \qquad (5)$$

Similarly, from clause (ii),

$$S_2 = S_2' + \alpha_2 \qquad (6)$$

We get from (iii), $S_3 = S_3' - \alpha_3$ (7)

Where $\alpha_1, \alpha_2, \alpha_3$ are non negative integer values because they all are equalizing factors in terms of bits.

It has already been stated that, $S_3 \rightarrow 0$ .

Consequently, $S_3' - \alpha_3$ will only tends towards zero if and only if, $\alpha_3 \rightarrow 0$ .

That is, it can be clearly deduced that,

$$\alpha_1 + \alpha_2 - \alpha_3 = c \geq 0 .$$

As a result, we may write,

$$\alpha_1 + \alpha_2 - \alpha_3 \geq 0 .$$

It has been previously stated that, $\alpha_3 \rightarrow 0$ .

Consequently, $\alpha_1 + \alpha_2 \geq \alpha_3$ (8)

Now, (5) + (6) + (7) results-

$$S_1 + S_2 + S_3 = S_1' + \alpha_1 + S_2' + \alpha_2 + S_3' - \alpha_3$$

$$\Rightarrow S_1 + S_2 + S_3 = S_1' + S_2' + S_3' + \alpha_1 + \alpha_2 - \alpha_3$$

$$\Rightarrow S_1 + S_2 + S_3 = S_1' + S_2' + S_3' + c \text{ [Using (8)]}$$

$$\Rightarrow S_1 + S_2 + S_3 \geq S_1' + S_2' + S_3'$$

$$\Rightarrow S \geq S' \text{ [Since } c \text{ is non-negative]}$$

That is, the performance of dictionary based data compression would be better for files with larger *TTR* Values and the result will deteriorate for lower *TTR* valued files. This is the aspect, which motivates us to provide peer files of each category to recognize the fluctuation of performance for dictionary based data compression.

## VII. Analysis and Discussions on the Proposed Corpus

In the previous sections, overview of the corpus has been presented. In this section, we present statistics regarding *TTR* and *Compression ratio* of the files of each groups. The experimental results demonstrate that, not only in expressing the effectiveness of any corpus, TTR may also be employed as a criteria for constructing data compressing corpus with the great scope of presenting the performance fluctuation of dictionary based (*i.e.* repetition analysis oriented) text compression scheme for best case and worse case analysis. The maximum *TTR* for each group is provided in Table IV.

TABLE IV

Maximum *TTR* For Each Group of Files of Proposed Corpus

| File Name | *TTR* |
|---|---|
| Article1 | 2.944 |
| Poem1 | 1.451 |
| Advertise1 | 1.170 |
| Speech1 | 5.846 |
| News1 | 3.087 |
| SMS1 | 1.138 |
| Email1 | 2.681 |
| Particulars1 | 3.510 |
| Story1 | 4.862 |
| Report1 | 3.364 |

The minimum *Type-to-Token Ratio* (*TTR)* for each group is provided in Table V.

TABLE V
MINIMUM *TTR* FOR EACH GROUP OF FILES OF PROPOSED CORPUS

| File Name | *TTR* |
|---|---|
| Article2 | 1.042 |
| Poem2 | 1.132 |
| Advertise2 | 1.012 |
| Speech2 | 2.164 |
| News2 | 1.611 |
| SMS2 | 1.103 |
| Email2 | 1.812 |
| Particulars2 | 1.489 |
| Story2 | 2.002 |
| Report2 | 2.157 |

We also analyze the compression ratio for the files using the dictionary-based approach for Bengali text compression provided in [7] and find the following statistics-

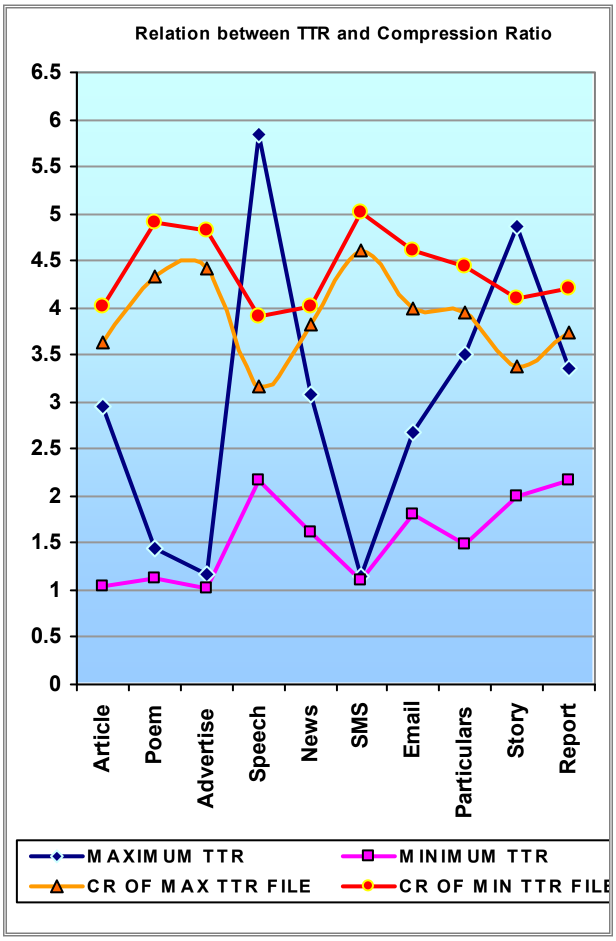

Figure. 2: Relation between Type to Token Ratio and Compression Ratio.

Fig 2, has presented a comparative analysis on Compression Ratio with *TTR* in terms of applicability of dictionary based data compression techniques. *Compression Ratio* (*CR*) is a metric of data compression efficiency. Compression ratio indicates the number of bits required to describe one byte in compressed form. The lower the compression ratio, the better is the compression. In order to evaluate the performance fluctuation of dictionary based Bengali text compression schemes, we propose to use the peer files with assumption that, the best case would be obtained from the files with greater *TTR* values and the worse case will occur for lower *TTR* files. The construction of the dictionary may really be facilitated from the files with larger *TTR*.

As, there is no existing corpus for evaluation of Bengali Text Compression Scheme, it is not possible to incorporate any comparative analysis with respect to any parameter. Because of the same in terms of compression benchmarks, there is no way to integrate the performance (effectiveness) analysis of proposed corpus. Basically, this is the reason which leads us to develop a new scheme for designing any corpus without any reference benchmarks. The proposed scheme is also to some extent a novel approach for automatic creation and evaluation of suitability (or unsuitability) of any corpora.

## VIII. APPLICABILITY OF PROPOSED CORPUS FORMATION SCHEME FOR OTHER LANGUAGES

The proposed method of building a text corpus for evaluation of Bengali text compression scheme is also applicable for non-Bengali data compression corpus formation. As per the presented scheme for building the corpus, we analyze various statistical analysis of the source text to decide whether the text will be included or not irrespective of the text-structure and other linguistic features. This text-structure independence feature provides a greater flexibility in applying the proposed concept for non-Bengali text.

We have already presented the theoretical background of choosing the peer files in view of dictionary based compression scheme. Here the same has been described in terms of traditional character based encoding scheme.

For a character based encoding scheme *i.e.* a single-gram based coding approach, the main strategy is to define code-words for each character and then replacing each character with corresponding code-word. The code-words are considered as binary stream and in practical cases the encoding scheme takes static coding into account. Static coding is an encoding mechanism where source components are encoded with non-conflicting variable length binary stream. In the simplest case, the length of each binary stream to be used for representing each character is determined through their frequency distribution. Often this length determination is obtained through probabilistic distribution based component ranking schemes.

Let there are *n* unit source symbols in the source language. That is, the total number of letters in the source language be *n*. If TTR is larger (greater) then the number (frequency) of matching word is also greater; consequently, number of distinct words is smaller. Whenever there will greater number of matching words, it may be assumed that, for average cases, the number of matching characters will also be greater. This may not be the same if the non-repeating words or low-repeating words individually contain unusually large set of repeating characters and consequently, even though the frequency of matching character is greater, sum of the characters present in the non-matching words is greater. That is, for a *n* character-set based language if the considering text consist of *m* characters, and there are $t_{mw}$ matching words where each of the matching word contains an average of $a_{cw}$ characters, the total pool of matching character is $t_{mw} * a_{cw}$ . Being *n* character-set based language, this pool of matching character may contain at best *n* distinct characters. That is, the TCR (Type to Character Ratio in analogy with TTR, Type to Token ratio) of the file will be a multiplier of $u = 1/n$ . Consequently, for any file with greater TTR, will have a large multiplier of *u*. As for practical cases we may assume that the average fluctuation of each word-length from average number of characters per word is approximately zero, we may consider the word length versus character distribution of the source text as a linear distribution. As a result, it may assume that, whenever there will have greater number of frequently occurring words, it implies frequently occurring characters.

Whenever static coding is used, the code-words are defined according to the frequency of the occurring elements. That is, frequently appearing elements will be assigned lower bit consuming codes in order to minimize the total overhead (in terms of bit consumption) whereas, the non-frequently occurring elements may be coded with greater threshold value. The same may be expressed in terms of Type to Character Ratio (TCR) that, elements with larger TCR will be assigned low overhead and elements with lower TCR will be assigned with grater overhead in comparison with lower TCR elements. If we want to deploy such scheme for employing in Bengali text compression scheme, it is an effective idea to build such a corpus which will contain reversible components, which presents the fluctuation of TCR clearly. That is, to provide a sound evaluation of the corpus, it may inherently express that, for a suitable corpus

$$t_{aw} \propto t_{ac} \text{ and,}$$

$$\frac{t_{aw}}{t_{dw}} \propto \frac{t_{ac}}{t_{dc}}$$

$$\frac{t_{aw}}{t_{dw}} \propto \frac{t_{ac}}{n}$$

Where,

$t_{aw}$ = Total number of words appeared in the source text.

$t_{dw}$ = Total number of distinct words in the source text.

$t_{ac}$ = Total number of characters appeared in the source text.

$t_{aw}$ = Total number of distinct characters in the source text.

*n* = Total number of symbols in the source language.

This criteria demonstrates an important and to some extent essential aspect of forming a data compression corpus that will facilitate of an effective and efficient design and implementation scheme of text compression approaches irrespective of language. Though the main concern or uses of data compression corpus that has been adapted traditionally is only evaluation of compression scheme, the evolving nature of data management has motivated (and necessarily forced) to develop corpora for being considered as a knowledgebase as well as test-base for text compression scheme. The analysis presented in this paper establishes the criteria to be taken into consideration for building any dictionary based text compression scheme irrespective of adapting single-gram dictionary based text compression or multi-gram dictionary based compression.

## VIII. Conclusion

We have proposed a new Corpus for evaluation of Bengali text compression schemes named *Ekushey-Khul*. The methodology is also robust to some extent that, it takes both linguistic and compression-ratio into account. It is also a contribution that, we specify new criteria for choosing text compression corpus, which includes consideration of expected device and the changed attitude of text varying with communication framework for which the compression is intended. Though it is an inaugurating step towards forming a Bengali text compression evaluation corpus, further improvement may be achieved by integrating other types and modes of document like technical documents, Bengali dictionary, and Bengali spreadsheets etc. with consideration of more training files.

## References

[1] Ross Arnold, and Tim Bell, “A Corpus for the evaluation of lossless compression algorithms”, *Data Compression Conference*, pp. 201-210, IEEE Computer Society Press, 1997.

[2] Khair Md. Yeasir Arafat Majumder, Md. Zahurul Islam, and Majuzmder Khan, “Analysis of and Observations from a Bangla News Corpus”, Proceedings of *9th International Conference on Computer and Information technology ICCIT 2006*, pp. 520-525, 2006.

[3] Mat Powel, “Evaluating Lossless Compression Algorithms”, February, 2001.

[4] N. S. Dash, "Corpus Linguistics and Language Technology*",* 2005.

[5] Official Web site of Prothom-Alo, www.prothom-alo.com

[6] M. Burrows and D. J .Wheeler. "A block sorting lossless data compression algorithm*".* Technical report, Digital Equipment Corporation, Palo Alto, CA, 1994.

[7] S. A. Ahsan Rajon, "A study on Bengali Text Compression Schemes", *Research Report*, Khulna University, Khulna, June 2008.

[8] Md. Rafiqul Islam, and S. A. Ahsan Rajon, "On the Design of an Effective Corpus for Evaluation of Bengali Text Compression Schemes", *Proceedings of 11th International Conference on Computer and Information Technology (ICCIT 2008),* 25-27 December, 2008, Khulna, Bangladesh, pp. 236-241.

[9] Niladri Sekhar Dash, "Some Techniques Used for Processing Bengali Corpus to Meet New Demands of Linguistics and Language Technology", *SKASE Journal of Theoretical Linguistics*. 2007, vol. 4, no. 2 ISSN 1336-782X.

[10] Avik Sarkar, Anne De Roeck , A Framework for Evaluating the Suitability of Non-English Corpora for Language Engineering [online].

[11] Akshar Bharathi, Rajeev Sangal and Sushma M Bendre: Some Observations Regarding Corpora of Indian Languages. Proc. of Int. Conf. on Knowledge-Based Computer Systems (KBCS-98), 17-19 Dec 1998, NCST, Mumbai.

[12] Niladri Sekhar Dash and Bidyut Baran Chaudhuri , Why do we need to develop corpora in indian languages. [online]

[13] Upal Garain, B. B. Chaudhuri, "A Complete printed Bangla OCR System", Pattern Recognition Journal, Vol. 31, No. 5, pp. 531 – 549, Elseiver Science Limited, 1998.

**Md. Rafiqul Islam** obtained Master of Science (M. S.) in Engineering (Computers) from Azerbaijan Polytechnic Institute (Azerbaijan Technical University at present) in 1987 and Ph.D. in Computer Science from Universiti Teknologi Malaysia (UTM) in 1999. His research areas include design and analysis of algorithms and Information Security. Dr. Islam has got a number of papers related to these areas published in national and international journals as well as in referred conference proceedings.

He is currently working as the Head of Computer Science and Engineering Discipline, Khulna University, Bangladesh

**S. A. Ahsan Rajon** is an Adjunct Faculty of Computer Science and Engineering Discipline, Khulna University, Khulna. He has completed his graduation from the same discipline in April 2008. He is also pursuing M.B.A. from Business Administration Discipline under Management and Business Administration School of same university. Rajon has made several publications in International conferences. His research interest includes data engineering and management, electronic commerce and ubiquitous computing. Currently he is working on robotics.

He is a member of Institute of Engineers, Bangladesh (IEB).