Information Technology Journal 3 (1): 93-94, 2004
ISSN 1682-6027

A Comparative Study of Bangla Text Compression with Winzip

Md. Sazzad Hossain and [1]R.C. Debnath
Department of Physics, Rajshahi University of Engineering and Technology, Rajshahi-6204, Bangladesh
[1]Department of Applied Physics and Electronics, Rajshahi University, Rajshahi-6205, Rajshahi, Bangladesh

Abstract: This paper presents a comparative study of Bangla text compression with WinZip file. For better compression static Huffman technique is used in this compression. In this technique frequency distribution of the characters plays an important role for appropriate coding. These distributions are collected from different Bangla newspapers of different dated. Performance of this technique of Bangla text compression and Win Zipping technique for different lengths has been considered in terms of compression efficiency, coding and decoding times. The transmission cost of this proposed work is minimum than that of WinZip, although the proposed technique takes a little more time, which is not important factor than saving memory space due to compression.

Key words: Input from file, creatlinklist, sortlinklist, creattree, transfertextcontrol, merged

INTRODUCTION

Now-a-days without use of telex, teleprinter, fax and computer no nation can advance. To use Bangla in all spheres of life it is important to be able to adapt Bangla for use in modern technologies[1]. This language is being used for transmission of information from one town to another using computers. In these applications frequency distribution of Bangla text for efficient coding[2] static Huffman coding technique[3,4] is being used and as a result both the length of the transmitted code[5] and redundancy[6] have increased. Under these circumstances it has become very important to be able to compress Bangla text for improving efficiency of both storing and transmitting Bangla texts than Win Zipp.

MATERIALS AND METHODS

Procedure: Bangla text message is written in Microsoft word or WordPad that will be considered as input of original text message. After modeling or processing this message technologies[1] it is saved in unformatted system with extension text-document and then this unformatted text is compressed by using compression program that will be treated as compressed output file as Fig. 1. For decompression this compressed file taken as input and is decompressed by using decompression program and after processing we can get original Bangla document as output as Fig. 1.

Compression function: The compression program compresses the Bangla message with the help of its member functions. The function input_from_file( ) reads the data or message from input text file and links and sorts them by the functions CreateLinklist( ) and SortLinklist( ), respectively. The function CreateTree() creates tree of message. The function TransferTextControl( ) places the binary codes (from tree) according to input message. In this function encoded binary string is transferred into encoded characters. The function Merged( ) holds the tree and encoded characters of the message. The flowchart of the compression function (Fig. 2).

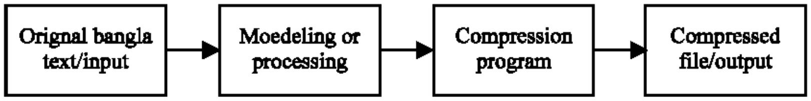

Fig. 1a: Block diagram of compression procedure

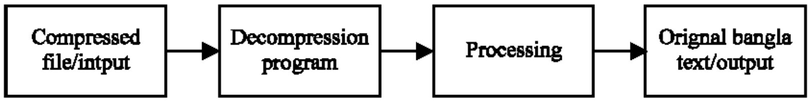

Fig. 1b: Block diagram of decompression procedure

Corresponding Author: Md. Sazzad Hossain, Department of Physics, Rajshahi University of Engineering and Technology, Rajshahi-6204, Bangladesh E-mail: sazzad_phy@yahoo.com

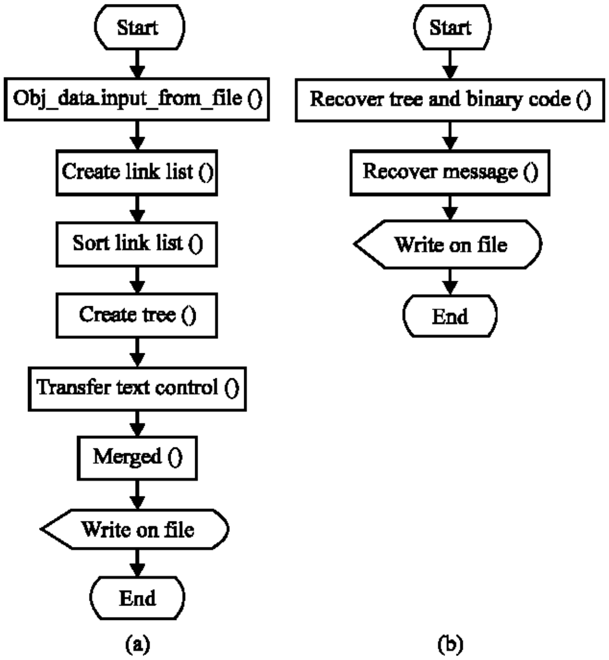

Fig. 2: The flowchart of the Bangla text (a) compression function and (b) decompression function

Decompression function: The decompression program decompresses or expands the Bangla message with the help of its member functions. The function Recover Tree And Binary Code( ) recovers tree and binary codes from encoded characters. The function Recover Message( ) recovers the text message with the help of tree. The flowchart of the compression function (Fig. 2).

RESULTS AND DISCUSSION

In this study we have to make compression files from Bangla.doc file. Practically the compressed file cannot be made from Bangla.doc directly. Bangla.doc is converted to text.doc (it is processed) first and then to compress. In that, it is simply a compression program. On the other hand, the software used for zipping is one kind of compression work. So, it has been compared between these two. Comparative results are given in Table 1.

In Table 1, Bangla.doc file, Rum41, file size is 30 kb taken as input file. After processing or modeling this file it is saved with extension text-doc in unformatted system and then compressed. We get the compressed file (Encd4) size is 3.34 kb. Compression efficiency is then 88.87%.

Table 1: Results of comparative study between Winzip and produced Bangla text compressed Files

File name and size		Win zipping		Comp. file name and size		Compre -ssion efficiency	Win zipping efficiency
Name.doc	Size	Name	Size	Name	Size	(%)	(%)
Rum41.doc	30.00kb	Rum41	5.75kb	Encd4	3.34kb	88.87	80.83
Rum31.doc	30.50kb	Rum31	5.92kb	Encd3	3.45kb	88.69	80.59
Rum11.doc	33.50kb	Rum11	6.77kb	Encd1	4.10kb	89.76	79.79
Rom11.doc	43.00kb	Rom11	8.89kb	Encd	6.17kb	85.65	79.33
Rom21.doc	67.50kb	Rom21	11.30kb	Encd2	10.90kb	83.85	83.26
BAN.doc	19.00kb	BAN	1.76kb	Entst	0.123kb	99.35	91.05

Using WinZip compression system for same input, compressed file size (Rum41) size is 5.75kb and efficiency is 80.83%. Similarly, considering the rest Bangla.doc file (Rum31, Rum11, Rom11, Rom21 and BAN) we have obtained by using Bangla text compression system and Win Zipping compression system efficiencies are 88.69, 89.76, 85.65, 83.85, 99.35% and 80.59, 79.79, 79.33, 83.26, 91.05%, respectively. In these results we see that the Bangla text compression is more efficient (about 8%) than the Win Zipping compression.

The result of our developed software is about more than 8% efficient than WinZip. Although its compressing and decompressing times a little more but compression efficiency is better than winZip. We firmly hope that the people will be benefited who uses this program for better compression and saving transmission cost.

REFERENCES

1. Bookestein, A. and J.A. Storer, 1992. Data Compression, Inform. Process. Manage., 28: 675-680.
2. Roy, C.K., M.M. Masud, M.M. Asaduzzaman and H.M. Hasan, 2001. A Modification of Huffman Header, ISBN 984-32-015-2, pp: 62-65.
3. Humayun, S.M., S.H. Rahman and M. Kaykobad, 1990. Static Huffman Code for Bangla Text, 15th Annual Conference of BAAS, Section III, AERE, Savar, pp: 5-8.
4. Mannan, M.A., R.A. Chowdhury and M. Kaykobad, 2001. A Storage Efficient Header for Huffman Coding, ICCIT, ISBN 984-32-015-2, pp: 57-59.
5. Cover, T.M., 1991. On the Competitive Optimality of Huffman Codes, IEEE Trans. Inform. Theory, IT-37, 1: 172-174.
6. Welch, T.A., 1984. A Technique for High-Performance Data Compression, Computer, 17: 8-19.