An analytical approach to Data Compression

Shubham Malik

(Reseach Scholar, NIET Greater Noida)

Abstract— Data compression is being most attractive , because it compresses data. This paper describes the two stages of encoding technique which compresses the data more efficient. This research paper provides a way to enhance the compression technique by merging incremental compression technique and RLE compression technique. In the phase one incremental compression algorithm stores the prefix of preceding symbol from the current symbol and replaces with numeric value. In phase two the data is compressed by (Run length encoding) algorithm that compresses the reoccuring data bits by short bits. This proposed technique increases the compression rate as from incremental compression technique and Run length encoding compression technique.

Index Terms—Incremental & Run length encoding, Compression, Compression technique, Enhancement of Incremental algorithm, Incremental encoding, Lossless compression, Lossy compression

1 INTRODUCTION

Data compression is way by which we decrease the size of input data and by the help of data compression the same data is stored in less bits. Data compression is also known as reverse decoding, and this generally encompasses the special representation of data which satisfies the need.With the help of this paper we will study and make algoritm for reducing the bits of the data .The main objective of data compression is to reduce the size of data that is to be send .By the help of this paper we will see different examples of data compression

Compression is very useful because it help to reduce the size of data with the help of that our transmission time and storage space is reduced .Data compression has important application in the area of file storage which manages the cost as well.

Now a day many application need large amount of data and day by day applications are built by which stores large amount of data.So that we can make our communication with the help of network or we can share the useful things like (audio ,vedio & text) with the help of network .If we send the original data it consume lot of space and also take large amount of time to transfer it with the help of data compression we minimize the time and space both .And performance of the system is also increased.

2 COMPRESSION TECHNIQUES

Compression techniques is divided into two parts of compression algorithm classes.

- ➢ Lossy compression algorithms

- ➢ Lossless compression algorithms

In compression algorithm input data is taken after that compression is applied to it .If someone want data same as original data then there are decompression algorithms as well .Compression varies from user to user how much does user want to compress it accordingly the compression is applied to the data

2.1 Lossless Compression

Lossless compression is a method in which the size of the input data is reduced but the meaning remain the same as the original data was . where no information is lost in compression that's why it is called as lossless compression as its name specify lossless means where no data is lost
For example: - lossless compression is very useful in vedio and text.

Because reconstruction of text message as the original message is very important, a small change in the message can change the meaning of the message. For example "DO NOT SMOKE HERE" and "DON'T SMOKE ".

If we are compressing the text it doesn't mean that the meaning is also changed .Then there is no benefit of compression .After compression the meaning of the text should remain the same then only there is benefit of compression is there .

The Zip file format is also the part of lossless compression data files .In this lossless compression is used .In lossless compression we want original data at output related to the input data otherwise we can use lossy compression .

Input data → compression → output data

2.2 Lossy Compression

As we can understand by its name 'lossy 'means data may be lost be lost or may not be .In lossy compression when we compression the data there is no surety that we will get the same or original data .some of our data may be lost .But the lost data can be retrieved if it is kept somewhere so that we can transfer our data easily .It depends on situation which type of data it is .If data is not required as exact as it was then we can use lossy compression as we can use it for audio, vedio, speech etc .This type of compression is used where loss of information is affordable

For example: there are some areas where exact original data is not required. During transmitting some speech and video, the data are not required as exact as original. They decompress according to the quality of the data required. This type of compression is used where any loss of information is affordable.
Lossy compression is mostly used for multimedia data (audio, video and images).In such type of data there is only variation in the colour intensity and pixel .That does not produce any difference in the image or vedio .We can understand the meaning from it .while lossless compression is applied where we want same point to point data.

According to the need of the user it can choose between these option which type of compression does he/she want to apply .If they want more compression ratio is not bothered about the loss then they can apply lossy other wise lossless is applied

Input data → compression → output data (data may not be same)

3 RUN LENGTH ENCODING

Run Length Encoding is widely used technique for data compression by which compression of data is done .Data some time contain same type of character or numeric value occuring many time by the help of RLE We can compress it. By which we can reduced the size of data and transfer time as well.

Let us understand this with the help of the example we take character as "sssss". Then output of the given character will be 5's' by the help of this we can reduce the size of the data.

3.1 Implementation

RLE is very easy to implement .Mainly it contain the physical size of the data that have same characters . It works like this way count the no of same character and place them accordingly first the numeric value and followed by character .If there is space character in the data then it is left as it is .It is useful where there is repeted data.

It contain two parts one is counter and other is text .where counter replace sequence of the same data value with in a file by a counter number and a single value where counter is then subtracted by 1because after the numeric value we write the symbol that's why 1 is subtracted here .counter value varies from 1 to 128 or 256 . we will understand it by the help of example

 Input- sssssssss

Output-10s

Example-sssssmmmmcccc

Output-5s4m4c

Here in the input we took 13 bytes which is to long but after applying the RLE algorithm it can be stored in 6 bytes .it give around 50% of compression ratio.

RLE algorithm works very quickly and requires less power of cpu work.rle algorithm work when there is large

redundant data.by the help of this we cannot achieve the highest ratio of data compression.

4 INCREMENTAL ENCODING ALGORITHM

Incremental algorithm stores the common data from front and back which help to remove the duplicate elements or data .

In others words we can say suffix or prefix is not repeated again rest part is printed with the previous output .Let us understand it by taking an example.

Input	Common prefix	Compressed output
to	No preceding word	0 to
too	`to `	2o
tooshu	`too `	3shu
cold	No common prefix	0 cold
colder	`cold`	4er
coldest	`cold`	4est
cold	`cold`	4

Table 1: Results of Incremental algorithm

The incremental encoding stores the common substring length that varies from application to application .Here we store a single byte which stores the same data .It is used for the sorted data .It may be combined with. other general lossless data compression technique

5 PRESENT TECHNIQUE

In present technique firstly we apply RLE compression

algorithm on data and after that we apply incremental compression algorithm on the result of RLE algorithm .We are using an escape character for identification between RLE compressed data and Incremental compressed data.

5.1 Comparison Analysis

With the help of example we will try to understand present technique and its effect in terms of compression ratio with RLE encoding & Increment Encoding .From which we create difference between both the algorithm . Now with the help of example we will understand it.

Below is the comparison table

Input Word	RLE Output	Incremental Output	Proposed Technique output
AAAAABBBCC CDDE	5A3B3CD DE	AAAAABBB CCCDDE	5A3B3CD DE
AAAAABBBCD	5A3BCD	8CD	9.D
AAAAABBCC CDD	5ABB3C DD	7CCCDD	7.3CDD
Takes 36 bytes	Takes 23 bytes	Takes 23 bytes	Takes 18 bytes
Compression%	**33%**	**33%**	**50%**

Table 2: Comparison analysis table

Here we have seen that uncompressed data take 36 bytes and RLE took 23 bytes same bytes by Incremental algorithm and in combination of both it took 18 bytes which is around 50% of compression in this present technique.

5.2 Implementation

Here we have implemented the present technique in two process which are

Ist process−We are applying RLE encoding technique to comprees the input data.

IInd process−We apply Incremental encoding technique on the output of RLE encoding technique.

INPUT DATA → RLE ALGORITHM → INCREMENTAL ENCODING ALGORITHM → OUTPUT DATA

The present representation of my research is

Step 1 - Take the first character from input string.

Step 2 - Write the first character to the output string.

Step 3 – Count the number of subsequent occurrence of the same string we have taken.

Step 4 – Write the counted number along with string to the output.

Step 5 – Take next character follow the same step again.

Step 6 – Repeat the same steps until our input data is traversed.

Step 7 – Print the final output of the result.

After applying the RLE algorithm we apply Incremental encoding algorithm to produce best result the output of the RLE works as the input to the incremental encoding algorithm . Now we see the steps of incremental algorithm which work after RLE.

Step 1 – Here we take input which is output of the RLE algorithm.

Step 2 – It count all the input string which are repeated and save it in output.

Step 3 – When another string comes to Incremental algorithm it compare with its output string and write it in its output.

Step 4 – Then for next word it compares with output and

write only the common characters and rest as it is.

Step 5 – repeat these step for all the string.

Step 6 – Print the final result.

6 KEY FEATURES

1. This is lossless compression technique .

2. Compression ratio is more than 50% in lossless compression.

3 . Speed up the transfer rate of data which is to be sent.

4 . Reduces the storage space where data is stored .

5 . Data remain unchanged .

6 . Cost effective.

7 CONCLUSION

The method gives very good result of data compression among the lossless compression .By the help of this method we can reduce the size of the data by 50%.We have discussed before with the help of example and found that in each case number of bytes used there . Among them the largest byte took by the Run Length Encoding after that second place taken by Incremental Encoding Algorithm and at last we applied combination of both then we get the desired output which compressed the data into half bytes which was taken by the other algorithms . Combination of both algorithm gives good compression ratio.

REFERENCES

1) http://en.wikibooks.org/wiki/Data_Compression

2) http://en.wikipedia.org/wiki/Data_compression

3) Alen H.writen ,Alistair Moffat ,Timothy C.Bell Managing gigabytes.

4) http://datacompression.info/

5) http://www.data-compression.com/index.shtml

6) http://www.webopedia.com/TERM/D/data_compression.html

7) http://www.arturocampos.com/ac_rle.html

8) http://www.binaryessence.com/dct/en000045.htm

9) http://www.tutorialcup.com

10) http://book.google.co.in