

MINOR PROJECT 1

SYNOPSIS

ON

Implementation and Performance Analysis of Different Compression Techniques

Submitted By

RAJUL DUBEY

50069424

DEVESH YADAV

500069565

KARAN SHARMA

500070100

SHRIYASH KAUSHIK

500067485

Under the guidance of

Dr. Hitesh Kumar Sharma

Associate Professor

Cybernetics Cluster

School of Computer Science



Department of Cybernetics,

School of Computer Science

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES

Dehradun-248007

Project Proposal Approval Form (2021-2022)

I

Minor

Group: 2

PROJECT TITLE:

**Implementation and Performance Analysis of Different
Compression Techniques**

TABLE OF CONTENT

1.	Abstract	4
2.	Introduction	5
3.	Objectives	7
4.	Methodology	8
5.	Tools and Technologies	9
6.	Pert Chart	10
7.	References	11

Abstract

Data compression is a common requirement for most of the computerized applications. Data compression is known for reducing storage and communication costs. , with data compression can shorten the time of data exchange. There are number of data compression algorithms, which are dedicated to compress different data formats. Even for a single data type there are number of different compression algorithms, which use different approaches. This project examines lossless data compression algorithms and compares their performance. A set of selected algorithms are examined and implemented to evaluate the performance in compressing text data. A data compression algorithm is to be developed which consumes less time while provides more compression ratio as compared to existing techniques.

Keywords: Text Data Compression, Huffman Algorithm, Shannon Fano Algorithm, Run Length Algorithm, lossless data compression.

Introduction

In this modern era of technology, everything is comprised of data. So, the problem in hand is to make the use of an adaptive compression technique, which can improve the compression ratio of existing compression techniques and could enhance the storage capacity of various storage devices by compressing the size of the original text file in a lossless manner. Data Compression Algorithms aim at minimizing the size of the data so that it occupies less amount of disk space and even help in reducing the network congestion since compressed data would use less bandwidth while being transmitted over a data communication channel.

Text compression aims at substituting the original symbol with a shorter symbol in the source code, which contains the same information but with a smaller system. In other words, using the data compression, the size of a particular file can be reduced. This is very useful when processing, storing or transferring a huge file, which needs lots of resources. If the algorithms used to encrypt works properly, there should be a significant difference between the original file and the compressed file. When data compression is used in a data transmission application, speed is the primary goal. Speed of transmission depends upon the number of bits sent, the time required for the encoder to generate the coded message and the time required for the decoder to recover the original ensemble. In a data storage application, the degree of compression is the primary concern. Compression can be classified as either lossy or lossless. Lossless compression techniques reconstruct the original data from the compressed file without any loss of data. Thus the information does not change during the compression and decompression processes. These kinds of compression algorithms are called reversible compressions since the original message is reconstructed by the decompression process. Lossless compression techniques are used to compress medical images, text and images preserved for legal reasons, computer executable file and so on. The decompression process results an approximate reconstruction. It may be desirable, when data of some ranges which could not recognized by the human brain can be neglected.

Problem statement

- 1) Data compression systems include as an integral part a certain classifier that allows you to select the most efficient compression algorithm.
- 2) A compression problem from the algorithmic point of view is to find an effective and efficient algorithm to remove various redundancy from certain types of data by using lossless data compression algorithm.
- 3) It is used to examines the performance of the Run Length Encoding Algorithm, Huffman Encoding Algorithm, Shannon Fano Algorithm.
- 4) In order to examine the performance of these algorithms in compressing text data is evaluated and compared.

Literature review

The problem of data compression is one of the important aspects in the development of information technology. Data compression is a process of resizing a file or document to be smaller in size. Along with the development of hardware and software technology is increasingly sophisticated and complex that demands the efficiency in terms of data storage and memory [1].

Communication is the exchange of thoughts, messages, or information, as by speech, visuals, signals, writing or behaviour. When two entities are communicating and do not want a third party to listen in or know, they need to communicate in such a way that it doesn't get intercepted. Secure communication is needed and there exists many tools for this. With secure communication, if we use compressed form of the messages which are being sent, this will make an effective and powerful system. Compressed message will be smaller in size than the original and less bits will be needed to make it confidential [2].

This is very useful when processing, storing or transferring a huge file, which needs lots of resources. If the algorithms used to encrypt works properly, there should be a significant difference between the original file and the compressed file. When data compression is used in a data transmission application, speed is the primary goal. Speed of transmission depends upon the number of bits sent, the time required for the encoder to generate the coded message and the time required for the decoder to recover the original ensemble. In a data storage application, the degree of compression is the primary concern. Compression can be classified as either lossy or lossless. Lossless compression techniques reconstruct the original data from the compressed file without any loss of data. Thus, the information does not change during the compression and decompression processes [3].

We require the ability to reconstitute the original file from the compressed version at any time. Data compression is a method of encoding rules that allows substantial reduction in the total number of bits to store or transmit a file. The more information being dealt with, the more it costs in terms of storage and transmission costs [4].

We check the performance of these three lossless data compression algorithms for compressing text data and then the compressed text data will be evaluated and compared.

Objectives

- To compare & evaluate the performance of the algorithms.
- To implement and compare the compressed file sizes using Huffman Encoding, Run Length Encoding, and Shannon Fano Algorithm.
- To reduce the amount of redundant information in the stored or communicated data.
- To compress the text data and also allows the original data to be reconstructed from the compressed data.

Methodology

There are number of different data compression methodologies, which are used to compress different data formats like text, video, audio, image files. Data compression techniques can be broadly classified into two major categories, “lossy” and “lossless” data compression techniques.

Currently there are many methods that can be used to compress data. And each method has different results and ways. In this report we will discuss the comparison of data compression using three different algorithms, there are using Shannon-Fano Algorithm, Huffman Algorithm, and Run Length Encoding Algorithm[5].

These are the following comparisons factors that have been selected:

1) **Compression Ratio** is the ratio between the size of the compressed file and the size of the source file.

$$\text{Compression Ratio} = \text{size after compression} / \text{size before compression}.$$

2) **Compression Factor** is the inverse of the compression ratio. That is the ratio between the size of the source file and the size of the compressed file.

$$\text{Compression Ratio} = \text{size before compression} / \text{size after compression}.$$

In order to test the performance of lossless compression algorithms, the Run Length Encoding Algorithm, Huffman Encoding Algorithm, Shannon Fano Algorithm are implemented and tested with a set of text files. Performances are evaluated by computing the above-mentioned factors.

- 1) Compression ratio using Huffman encoding.
- 2) Compression ratio using Run Length encoding.
- 3) Compression ratio using Shannon-Fano encoding

Tools and Technologies

System Requirements: (Software/Hardware)

1) Software requirements

1.1) Language: C

1.2) Operating system: Windows 7,8,10 or Mac

1.3) Compiler: GCC

2) Hardware requirements

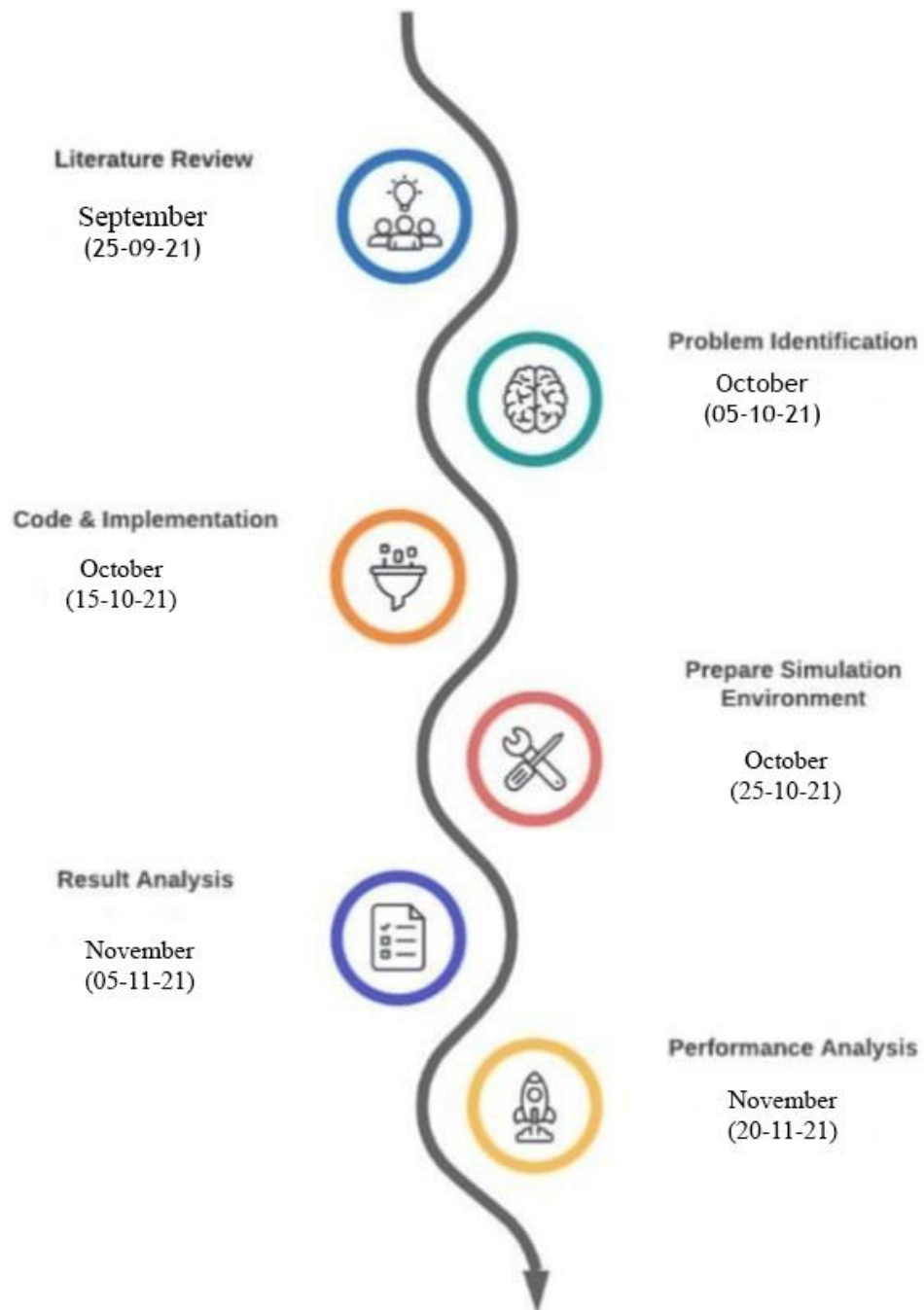
2.1) RAM: 512MB or above

2.2) Processor: Intel platinum

2.3) Hard Disk Size: 16GB for 32-bit OS or 20GB for 64-bit OS

Pert Chart [6]

Implementation and Performance Analysis of Different Compression Techniques



References

- [1] Dea Ayu Rachesti, Tito Waluyo Purboyo, and Anggunmeka Luhur Prasasti, “of Text Data Compression Using Huffman, Shannon-Fano, Run Length Encoding, and Tunstall Methods” ,available at <http://www.ripublication.com>. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 23 (2017) pp. 13618-13622.
- [2] Nigam Sangwan, ” Text Encryption with Huffman Compression”, International Journal of Computer Applications (0975 – 8887) Volume 54– No.6, September 2012, Institute of Technology and Management University, Gurgaon, Haryana, India.
- [3]S.R. KODITUWAKKU, & U. S.AMARASINGHE , ” COMPARISON OF LOSSLESS DATA COMPRESSION ALGORITHMS FOR TEXT DATA”, Indian Journal of Computer Science and Engineering Vol 1 No 4 416-426.
- [4] Amandeep Singh Sidhu, & Er. Meenakshi Garg, “Research Paper on Text Data Compression Algorithm using Hybrid Approach”, International Journal of Computer Science and Mobile Computing, Vol.3 Issue.12, December- 2014, pg. 01-10.
- [5] International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 5, Issue 6, June 2016
- [6] Pert Chart <https://my.visme.co/>

Approved By

Dr. Hitesh Kumar Sharma
Mentor

Dr. Monit Kapoor
Head Of Department



RAJUL DUBEY

Sir Synopsis After changes please check

6:53 PM (21 minutes ago) ☆



Dr. Hitesh Kumar Sharma

to me ▾

6:54 PM (19 minutes ago) ☆ ↩ ⋮

Approved.

You can go with this topic.

All the best team.

-Dr. Hitesh

From: RAJUL DUBEY <500069424@stu.upes.ac.in>

Sent: Saturday, September 25, 2021 6:53 PM

To: Dr. Hitesh Kumar Sharma <hksharma@ddn.upes.ac.in>

Subject: Re: Minor project Synopsis

...