Huffman coding for text compression

Presented by: Yash Rangra(545)
Harsh Rashinkar(546)
Abhay Sawant (547)
Suryansh Shambharkar(548)

Course : Digital Communication

Guided by : Suchitra Patil

Introduction

- Data compression reduces storage and transmission cost.
- Huffman coding is lossless and widely used.
- Assigns short codes to frequent symbols and long codes to rare symbols.
- Proposed by David A. Huffman in 1952.

Theory/Working

- Step 1: Calculate character frequencies.
- Step 2: Build Huffman tree using greedy approach.
- Step 3: Generate prefix-free binary codes.
- Step 4: Encode text into compressed bits.
- Step 5: Decode bits back into original text.

Application

- Used in ZIP, GZIP file compression.
- JPEG, PNG image compression.
- MP3, AAC audio compression.
- MPEG, H.264 video compression.
- SMS, Email, storage systems.

Result

- Original Size is 100%
- Compressed Size is smaller than ~40–60% depending on input.
- Entropy (H) is less than Average Length (L) → Efficient coding
- Efficiency is ~85–95%

Conclusion

- Uses variable-length binary codes based on character frequency.
- Provides lossless data compression with no information loss.
- Reduces file size and improves storage efficiency.
- Widely used in file, image, and data compression applications.