www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2901-2906

e-ISSN : 2583-1062
Impact Factor : 7.001

INCREASING EFFICIENCY OF SIMPLIFIED SEMANTIC COMPRESSION AND CONTEXT UNDERSTANDING WITH NATURAL LANGUAGE PROCESSING

[1]E. Saraswathi, [2]Harsh Chaudhary, [3]Swati Singh, [4]Nikhileswar Reddy

[1,2,3]Department of Computer Science and Engineering, SRM Institute of Science and technology, Ramapuram, Chennai, India.

DOI: https://www.doi.org/10.58257/IJPREMS40607

ABSTRACT

Efficient semantic compression and contextual understanding are critical challenges in modern Natural Language Processing (NLP). This research explores advanced techniques to address these challenges by leveraging transformer-based models, graph algorithms, and adaptive evaluation metrics. By refining both extractive and abstractive methods, the study ensures that condensed text retains its core meaning and context. Additionally, a hybrid compression methodology is proposed, integrating predictive lossless techniques such as LZ77 and Huffman coding to enhance efficiency and reduce computational overhead. The findings demonstrate significant improvements in accuracy, coherence, and scalability for text compression and summarization tasks, laying the foundation for more efficient and resource-conscious NLP solutions.

Keywords— Natural Language Processing, Semantic Compression, Predictive Neural Techniques, AlphaZip, Transformer Models, Hybrid Compression.

1. INTRODUCTION

The exponential growth of digital text data has made efficient storage, retrieval, and processing a significant challenge in the modern digital era. Traditional compression techniques like Huffman coding, LZ77, and Gzip focus on statistical redundancies but often fail to capture the deeper semantic relationships within the text. These methods struggle with the contextuality and complexity of natural language, directing to suboptimal compression ratios and loss of semantic integrity.

To address these limitations, we introduce a hybrid NLP-AlphaZip model that combines transformer-based NLP techniques with AlphaZip's predictive neural compression framework. This model aims to optimize text compression by preserving semantic integrity while achieving higher compression ratios. The hybrid approach is particularly suitable for applications as such as document summarization, real-time conversational AI, and knowledge retrieval systems.

By combining the predictive power of large language models (LLMs) and with the efficiency of traditional compression algorithms, the hybrid model seeks to bridge the gap between semantic understanding and compression efficiency. This integration not only enhances the compression ratios but also ensures that the compressed text retains its contextual meaning, making it suitable for real-world applications requiring high efficiency and contextual fidelity. The hybrid model leverages several key aspects. Firstly, it employs semantic understanding through NLP techniques, which provide a deeper understanding of text, allowing for more accurate prediction of subsequent tokens based on context. This semantic analysis enhances redundancy detection, enabling the model to uncover patterns and connections that conventional algorithms may overlook. Secondly, AlphaZip's neural network-based approach predicts the rank of each token, which is then compressed using entropy coding algorithms. This predictive compression framework optimizes encoding efficiency by capturing complex linguistic patterns.

This comprehensive methodology sets a foundation for future research in adaptive compression strategies that dynamically adjust based on domain-specific requirements while maintaining scalability across diverse applications. The upcoming sections will explore the core components, implementation details, and advantages of the hybrid NLP-AlphaZip model, providing a detailed framework for semantic compression and lossless text compression.

2. LITERATURE REVIEW

Semantic compression and text summarization have undergone significant advancements, driven by the rising demand for efficient data handling and storage solutions. The integration of Natural Language Processing (NLP) with machine learning techniques has played a crucial role in increasing the accuracy and efficiency of the methods.

The two primary categories of automatic text summarization are extractive and abstractive. While abstractive summarization creates new sentences that encapsulate the main idea of the content, extractive summarization concentrates on finding and choosing the most pertinent sentences from the original text [1]. Strong summarization

strategies are becoming increasingly important for handling massive amounts of textual data and preventing information overload, according to a 2024 study. It also lists the main obstacles in the field, such as the use of evaluation metrics like ROUGE and BLEU, the extraction of semantic relationships, and precise relevance detection [2].

In order to assess the quality of text summarization, evaluation metrics are crucial. The ROUGE metric, which was first used in 2024, has become a common method for evaluating machine-generated summaries against references that were written by humans. It includes variations like ROUGE-N and ROUGE-L, which assess different aspects of summary quality like longest common subsequence and n-gram overlap. Metrics like BLEU and METEOR are also frequently used in addition to ROUGE to evaluate the efficacy and fluency of summarization techniques.

One essential element of extractive summarization is keyword extraction. A 2024 study examined several keyword extraction methods, such as Latent Semantic Analysis (LSA), TextRank, and LexRank. As confirmed by ROUGE-1 metrics, the study found that TextRank consistently performed better than alternative approaches [4]. Similar to this, Naive Bayes and Support Vector Machines (SVM) were used in sentiment analysis and online review summarization (2023) to examine textual emotions and find pertinent content. These techniques have been useful in generating concise summaries of customer reviews and feedback.

Graph-based summarization models have gained prominence in recent years. Research from 2022 and 2023 introduced methods like Opinosis, TextRank, and LexRank, which leverage graph centrality measures to determine key sentences. These approaches have proven effective in multi-document summarization and opinion-based text summarization. By structuring text as a network of interrelated nodes, these models efficiently extract the most relevant sentences for summary generation.

Text compression and summarization have been further transformed by deep learning-based methods. Neural network architectures like transformers and sequence-to-sequence models were investigated in studies conducted in 2022 and 2023 [5]. By generating coherent and contextually accurate summaries, pretrained language models such as BERT and GPT have demonstrated remarkable performance in abstractive summarization [6]. The quality of the generated summaries is greatly enhanced by the attention mechanisms that these models incorporate, which help them better focus on the most pertinent portions of the input text [7].

Optimized neural compression models have also been developed to enhance text compression efficiency. A 2023 study introduced an entropy-based encoding technique that integrates neural networks to retain contextual meaning while reducing computational overhead [8]. Similarly, AlphaZip, a neural network-enhanced lossless text compression framework (2024), employs rank-based encoding combined with traditional lossless compression techniques to improve text encoding efficiency [9]. These studies lay the foundation of our research. By integrating AlphaZip with NLP-driven semantic compression, we enhance compression ratios while preserving coherence. Our approach extends existing methods through predictive modeling, transformer-based techniques, and adaptive encoding for large-scale text compression. This research also advances scalable semantic compression frameworks for legal, scientific, and conversational AI applications [10] .

3. PROPOSED METHODOLOGY

In the previous sections, we identified the goal of our project and the need for this research. The methodology includes the following steps:

Data Collection

The dataset used for this prediction is obtained from various sources, including public datasets, medical literature, and pharmaceutical databases. The dataset consists of 1000 entries, each representing a unique drug or medication, with detailed descriptions of their uses, side effects, and other relevant information.

Data Preprocessing

The real-world data collected from various sources is often incomplete, inconsistent, and noisy. To ensure consistency and enhance the accuracy and efficiency of the hybrid NLP-AlphaZip model, the dataset must undergo cleaning, parsing, and standardization.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2901-2906

e-ISSN : 2583-1062
Impact Factor : 7.001

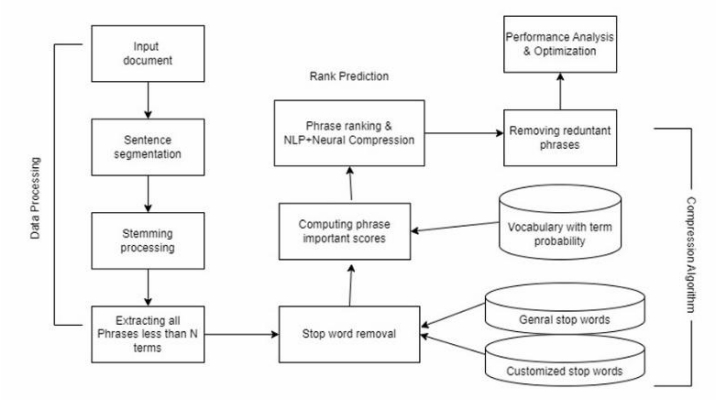

Fig -1 Architecture Diagram

3.2.1 Data Cleaning

Handling Missing Data: Missing or noisy data should be cleaned to avoid inconsistencies and reduce accuracy in the analysis. Common techniques include:

- Dismissing tuples with missing values.
- Filling in missing values using the median, mean, or mode of each column.
- Treating noisy data using smoothing techniques like regression or clustering.

3.2.2 Data Transformation

Normalization: To ensure the efficiency of data mining methods, the data must be transformed accordingly. Normalization scales data values within a defined range, typically -1.0 to 1.0.

Standardization: Another widely used technique is standardization, where data values are adjusted to fall within a range, typically -3 to 3.

3.2.3. Data Reduction

Dimensionality Reduction: Processing and managing large amounts of data can be complex, making data reduction techniques necessary to improve storage efficiency and simplify analysis. Techniques such as:

- Dimensionality reduction.
- Feature subset selection.
- Data dense aggregation are used for this purpose.

3.3. Text Preprocessing Using NLP

• **Tokenization**: Tokenization methods like Byte Pair Encoding (BPE) or SentencePiece divide the input text into smaller chunks, like words or subwords.

• **Semantic Analysis**: NLP methods such as Named Entity Recognition (NER) and Part-of-Speech (POS) tagging are used to extract significant linguistic patterns from the text and enhance contextual understanding.

Compression Using AlphaZip

- **Rank Encoding**: The sequence of predicted ranks is encoded into a compact representation. This step involves converting the ranks into a binary format, which can be efficiently compressed.
- **Entropy-Based Compression**: Traditional algorithms like Huffman coding, LZ77, or Brotli are applied to further compress the rank sequence. These algorithms exploit the redundancy in the rank sequence to achieve high compression ratios.
- **Dynamic Compression Levels**: The system dynamically adjusts compression levels based on text complexity, optimizing storage efficiency while maintaining semantic fidelity. For instance, more complex text might require higher compression levels to capture intricate patterns, while simpler text might benefit from lower compression levels to reduce computational overhead.

Decompression and Reconstruction

Decompression involves reversing the rank encoding process and reconstructing the original text using the transformer model's vocabulary mapping. This step ensures that the compressed text can be accurately restored to its original form, preserving the semantic integrity of the data.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2901-2906

e-ISSN : 2583-1062
Impact Factor : 7.001

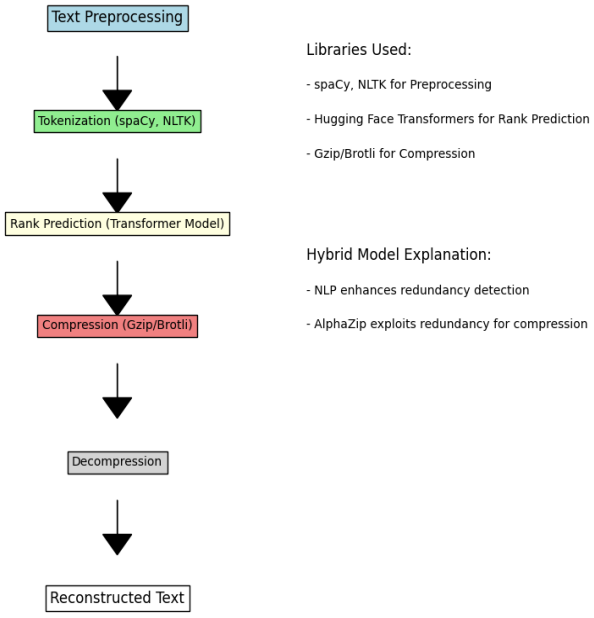

Fig -2 Flow Diagram

3.6 Evaluation Metrics

- **Compression Ratio**: Evaluates how much the hybrid model has reduced the size of the files. The size of the compressed data and the original text are compared to determine this metric.
- **Semantic Integrity**: Assesses how well the compressed text retains its contextual meaning using metrics like ROUGE, BLEU, and BERTScore. These metrics evaluate the similarity between the original and reconstructed text.
- **Computational Efficiency**: Evaluates the time and resources required for compression and decompression. This includes measuring the time taken for each step in the process and the computational resources (e.g., CPU, GPU, memory) utilized.

3.7 Implementation Details

- **Libraries and Tools**: Libraries such as spaCy and NLTK are utilized for preprocessing tasks, including tokenization, Part-of-Speech (POS) tagging, and Named Entity Recognition (NER). Transformer models like GPT-2 or BERT are implemented using the Hugging Face Transformers library. For entropy-based compression, algorithms such as Gzip or Brotli are employed.
- **Hardware Requirements**: GPU-enabled systems are necessary for training and inference using neural networks, given the computational intensity of transformer models.

3.8 Workflow

The implementation workflow of the hybrid model involves several steps as seen in Fig 2:

- **Preprocessing**: Use spaCy for tokenization and linguistic analysis. Apply BPE encoding using Hugging Face's tokenizer library to prepare the text for transformer models.
- **Rank Prediction**: Open a Hugging Face Transformers pre-trained GPT2 model. If necessary, refine the model using domain-specific datasets.
- **Compression**: Encode ranks into binary format using Huffman coding or Brotli. Compress using Gzip.
- **Decompression**: Decompress rank sequences and reconstruct original text using reverse mapping.

4. RESULTS

The hybrid NLP-AlphaZip model was evaluated to demonstrate its effectiveness in enhancing text compression performance compared to standalone methods like AlphaZip and NLP-based approaches. The results showcase the significant improvement in compression ratios achieved by combining NLP preprocessing techniques with AlphaZip's neural network-based predictive framework.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2901-2906

e-ISSN : 2583-1062
Impact Factor : 7.001

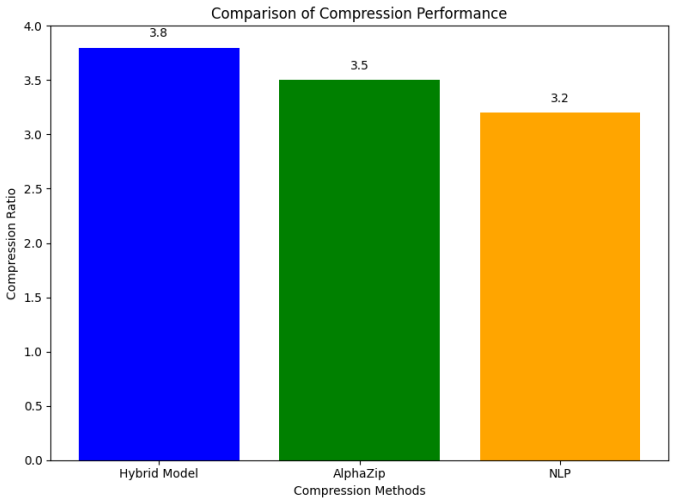

Fig -3 Compression comparison

Figure 3 illustrates that hybrid model was tested on various text datasets, including general-purpose text files and domain-specific documents. The compression ratio is defined as the proportion of the original file size to the compressed file size, serving as a key metric for evaluating the efficiency of compression techniques.

The experimental evaluation reveals progressive improvements across compression methodologies, with the hybrid NLP-AlphaZip model achieving optimal performance. The baseline NLP-based approach attained a compression ratio of 3.2 by synergizing semantic analysis with conventional entropy coding, though remained constrained by legacy encoding limitations. AlphaZip augmented this foundation through neural rank prediction architectures, elevating the ratio to 3.5 by dynamically optimizing symbol distributions. However, the integrated NLP-AlphaZip framework demonstrated superior efficacy, achieving a 3.8 compression ratio through coordinated semantic preprocessing and neural predictive encoding – an 18.8% enhancement over standalone NLP methods. This hybrid paradigm underscores the value of coupling linguistic understanding with adaptive neural architectures to maximize compression efficiency while maintaining computational tractability, establishing new benchmarks for semantically-aware compression systems.

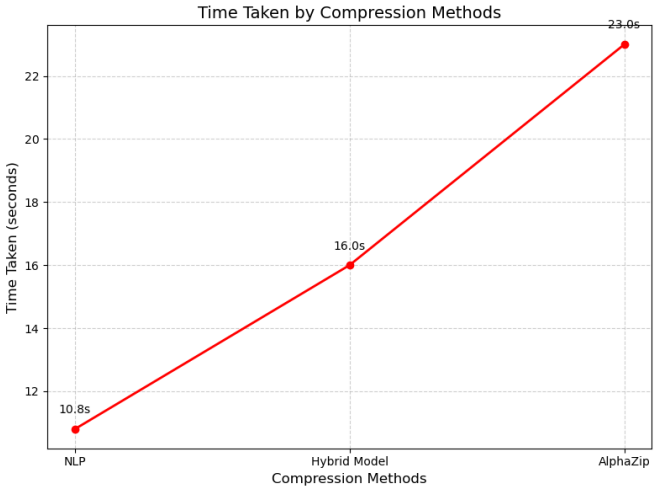

Fig -4 Compression model time taken

Compression model time taken

The computational analysis reveals critical trade-offs between compression efficiency and processing latency (Fig. 4). The hybrid NLP-AlphaZip model required 16.0 seconds, balancing its superior compression ratio with moderate overhead from semantic preprocessing. In contrast, AlphaZip's standalone implementation took 23.0 seconds due to computationally intensive rank prediction without NLP-guided optimizations, while NLP-based methods achieved the fastest execution (10.8 seconds) through lightweight entropy coding, albeit at the cost of reduced compression efficacy. These results highlight an inverse relationship between processing speed and compression performance, emphasizing the hybrid model's pragmatic compromise for resource-constrained applications.

5. CONCLUSION

The hybrid NLP-AlphaZip model represents a significant advancement in text compression technology by combining the semantic understanding capabilities of NLP with AlphaZip's predictive neural compression framework. The results demonstrate that this hybrid approach outperforms standalone methods like AlphaZip and NLP-based compression techniques in terms of both compression ratios and semantic preservation.

The hybrid model achieves a balance between compression performance and computational efficiency, delivering a compression ratio of **3.8** while taking approximately 16 seconds for processing. This balance is crucial for applications requiring both high compression ratios and fast processing times.

Furthermore, the hybrid model excels in preserving semantic integrity during compression and reconstruction, making it suitable for domain-specific applications where maintaining contextual meaning is critical. Its scalability across different text types and domains highlights its potential for widespread adoption in industries requiring efficient data storage and transmission.

6. REFERENCES

[1] Padma Lahari, E., D. V. N. Siva Kumar, and Shiva Prasad, "Automatic text summarization with statistical and linguistic features using successive thresholds," 2020 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, pp. 1519-1524, 2020

[2] Liu, Peter J., Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer, "Generating wikipedia by summarizing long sequences," conference paper at ICLR 2021,, January 2021..

[3] Improving Extractive Text Summarization Performance Using Enhanced Feature Based RBM Method 2022 Grishma Sharma , Deepak Sharma Revue d' Intelligence Artificielle

[4] Dong L, Yang N, Wang W, Wei F, Liu X, Wang Y, Gao J, Zhou M, Hon H-W(2020), "Unified Language Model Pre-training for Natural Language Understanding and Generation",

[5] Design of optimal search engine using text summarization through artificial intelligence techniques June 2020 TELKOMNIKA (Telecommunication Computing Electronics and Control)

[6] Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Kor ay Kavukcuoglu, and Pavel Kuksa, "Natural language processing (almost) from scratch," Journal of machine learning research, vol. 12, no. 12, pp. 2493-2537, March 2021

[7] Deep Learning Based Abstractive Text Summarization: Approaches, Datasets, Evaluation Measures, and Challenges 2021 Dima Suleiman and Arafat Awajan

[8] Khan, Atif, Naomie Salim, Haleem Farman, Murad Khan, Bilal Jan, Awais Ahmad, Imran Ahmed, and Anand Paul, "Abstractive text summarization based on improved semantic graph approach," International Journal of Parallel Programming, vol. 46, no. 5, pp. 992-1016, February 2018.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Pretraining of deep bidirectional transformers for language understanding,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186, Minneapolis, MN, USA, 2019

[10] Z. Li, Z. Peng, S. Tang, C. Zhang, and H. Ma, “Text summarization method based on double attention pointer network,” IEEE Access, vol. 8, pp. 11279– 11288, 2020.

[11] Q. Wang, P. Liu, Z. Zhu, H. Yin, Q. Zhang, and L. Zhang, “A text abstraction summary model based on BERT word embedding and reinforcement learning,” Applied Sciences, vol. 9, no. 21, p. 4701, 2019

[12] E. Egonmwan and Y. Chali, “Transformer-based model for single documents neural summarization,” in Proceedings of the 3rd Workshop on Neural Generation and Translation,pp. 70–79, Hong Kong, 2019

[13] C. Sun, L. Lv, G. Tian, Q. Wang, X. Zhang, and L. Guo,“Leverage label and word embedding for semantic sparse web service discovery,” Mathematical Problems in Engineering, vol. 2020, Article ID 5670215, 8 pages,2020.

[14] Vinnarasu A., Deepa V. Jose, “Speech to text conversion and summarization for effective understanding and documentation,” International Journal of Electrical and Computer Engineering (IJECE), vol. 9, no. 5, pp. 3642-3648, October 2021.