August 2023

Better Text Compression Using a Large Language Model

D Shin

Better Text Compression Using a Large Language Model

ABSTRACT

Conventional compression techniques for text are based on typical frequencies of individual letters within the text, independent of higher-level semantics. This disclosure describes a compression scheme for text data in which a conventional coder/decoder (codec) is augmented with an additional semantic codec to achieve greater compression and throughput. The additional semantic codec can be implemented with a pre-trained large language model (LLM). Text data is first input to a semantic coder for semantic-based compression. Codes within the codebook are reranked based on selective erasure by the encoder LLM. Once the codebook is established, portions within the text that can be recovered by a decoder LLM are erased. Such semantically compressed data is encoded as usual via conventional techniques and can be first decoded via conventional techniques to recover the semantically coded text. The semantically coded text is further decoded using a semantic decoder that recovers the original text by inferring, based on semantics, the portions that were erased prior to transmission.

KEYWORDS

- Semantic encoder
- Semantic decoder
- Large language model (LLM)
- Text recovery
- Text compression
- Huffman coding
- Token prediction

BACKGROUND

Data transfer between devices can involve sending large amounts of text (or information formatted as text, e.g., comma or tab-separated values) from one device to another. Such transfers can occur over any suitable channel, such as a direct connection between two devices or a network such as the Internet. To optimize data throughput and bandwidth use, text data is usually compressed prior to transmission by encoding it via conventional techniques, such as Huffman coding. At the receiving end, the incoming transmission is uncompressed via the corresponding decoding technique to recover the text data in the original form.

Humans often use semantics to complete sentences with missing words. For example, a human is highly likely to infer that the sentence "the boy walked [blank] school" is meant to be "the boy walked ***to*** school." However, the techniques for encoding and decoding the data are designed based on typical frequencies of individual letters within the text. As such, conventional compression techniques are independent of higher-level semantics of the text content even though such semantics can help achieve higher compression ratios by dropping the content that can be inferred with accuracy based on higher-level semantics.

DESCRIPTION

This disclosure describes a transmission scheme for text data in which a conventional coder/decoder (codec) is augmented with an additional semantic codec to achieve greater compression and throughput. The additional semantic codec can be implemented with a pre-trained large language model (LLM). A trained large language model is capable of high accuracy token prediction, e.g., words or phrases from a text that are omitted, and can therefore be utilized in text encoding/decoding to achieve higher compression ratios than conventional techniques.

The text data to be transmitted is first input to the semantic coder for semantic-based compression. Codes within the codebook are reranked based on selective erasure by the encoder LLM. Once the codebook is established, portions within the text that can be recovered by a decoder LLM at the receiving end are erased. Such semantically compressed data is encoded as usual via conventional techniques and transmitted to the receiving parties. At the receiving end, the encoded transmission is first decoded via the conventional techniques to recover the semantically coded text. The semantically coded text is further decoded using a semantic decoder that recovers the original text by inferring, based on semantics, the portions that were erased prior to transmission.

The operation of the semantic encoder can employ a prompt interface layer to the pre-trained LLM to generate the codebook by selectively erasing portions, such as specific words within sentences, in the original text. For example, the encoding prompt interface layer can be utilized to set the token history of the LLM to find optimal erasure across the entire input text. At the receiving device, the prompt interface layer to the LLM can be configured for semantic decoding to recover the original text by completing the erased words subsequent to conventional prefix decoding.

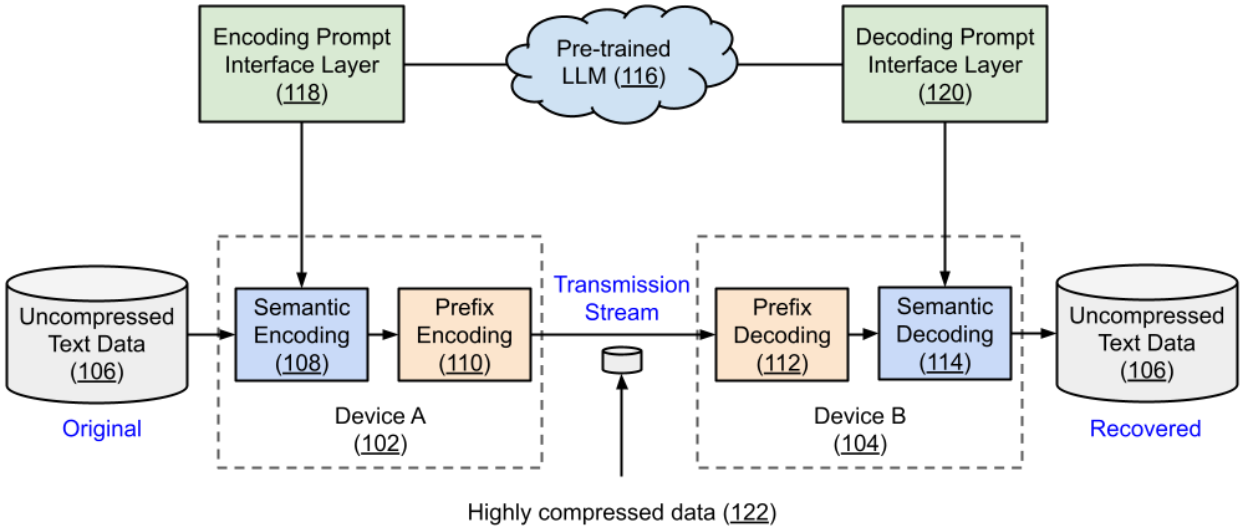

Fig. 1: LLM-assisted semantic encoding for higher compression ratio for text

Fig. 1 shows an example operational implementation of semantics-aware compression, per the techniques of this disclosure. A large volume of original, uncompressed text data (106) is to be transmitted from device A (102) to device B (104). The data is first semantically encoded (108) by employing an encoding prompt interface layer (118) to a pre-trained LLM (116). Next, prefix encoding (110) is performed using conventional encoding techniques to obtain highly compressed data (122) that is transmitted to device B.

At device B, the compressed data received as a transmission stream is first decoded by prefix decoding (112) to recover the semantically coded data. The semantically coded data is decoded (114) by employing a decoding prompt interface layer (120) to the pre-trained LLM (116) that was used for semantic encoding of the data. Semantic decoding results in recovering the original uncompressed data with a high degree of accuracy.

The techniques described in this disclosure can be easily implemented as an addition to any existing codec architectures that use any conventional encoding schemes. The techniques can incorporate any suitably pre-trained LLM as long as the LLM is common to the semantic

encoder at the sending end and the semantic decoder at the receiving end (or has similar predictive capability). The addition of semantic encoding as described herein can enable substantially higher compression rates for text, with little to no loss in codec accuracy. The increase in compression can enhance the throughput and efficiency of the transmission of text, thus lowering costs and boosting speed.

CONCLUSION

This disclosure describes a compression scheme for text data in which a conventional coder/decoder (codec) is augmented with an additional semantic codec to achieve greater compression and throughput. The additional semantic codec can be implemented with a pre-trained large language model (LLM). Text data is first input to a semantic coder for semantic-based compression. Codes within the codebook are reranked based on selective erasure by the encoder LLM. Once the codebook is established, portions within the text that can be recovered by a decoder LLM are erased. Such semantically compressed data is encoded as usual via conventional techniques and can be first decoded via conventional techniques to recover the semantically coded text. The semantically coded text is further decoded using a semantic decoder that recovers the original text by inferring, based on semantics, the portions that were erased prior to transmission.

REFERENCES

1. Zhang, Wenyu, Kaiyuan Bai, Sherali Zeadally, Haijun Zhang, Hua Shao, Hui Ma, and Victor Leung. "DeepMA: End-to-end deep multiple access for wireless image transmission in semantic communication." *arXiv preprint arXiv:2303.11543* (2023).
2. Xue, Ruiqing, Yanqing Liu, Lei He, Xu Tan, Linquan Liu, Edward Lin, and Sheng Zhao. "FoundationTTS: Text-to-Speech for ASR Customization with Generative Language Model." *arXiv preprint arXiv:2303.02939* (2023).
3. Pagé, Phillippe. "Abbreviated Semantic Encoding: A Technique for Prompt Compression." available online at https://medium.com/@philippeandrepage/abbreviated-semantic-encoding-a-technique-for-prompt-compression-394ea5ede381 accessed July 23, 2023.