Lossless Neural Text Compression

Kasey Luo, Michael Herrera

{kaseyluo, herrerx} @stanford.edu

Introduction

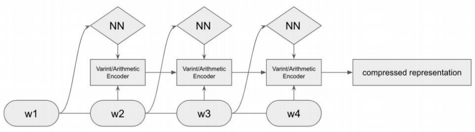
Problem + Research Question

- Data is being generated at a rapid rate. How can we more efficiently compress text data?
- Can transformers be leveraged to more efficiently compress text?

Background

- Existing RNN + encoding approaches for character-based text compression
- Existing NN approaches for image compression

Methods


NN
NN
NN
Varint/Arithmetic Encoder
Varint/Arithmetic Encoder
Varint/Arithmetic Encoder
compressed representation
w1
w2
w3
w4

Model:

- Pretrained GPT2 model, finetuned on text generation task on the wikitext2 dataset.
- Hyperparameters: learning rate of 2e-05, training and evaluation batch size of 8, Adam optimizer

Variable-Length Integer Encoding

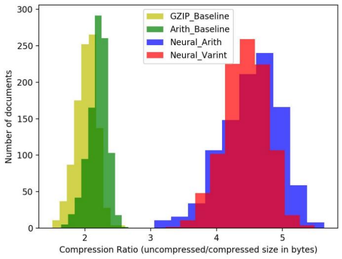
- Compresses fixed-length integers into variable-length integers by storing smaller numbers with fewer bits

Arithmetic Encoding

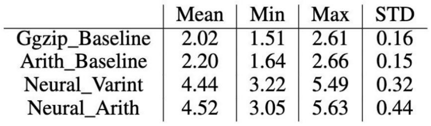
- Frequently used characters are stored with fewer bits and less frequent characters stored with more bits

Experiments & Results

Compression Ratio Histogram (Uncompressed/Compressed)

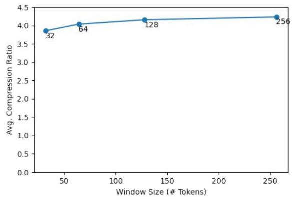


Compression Ratio Table (Uncompressed/Compressed)

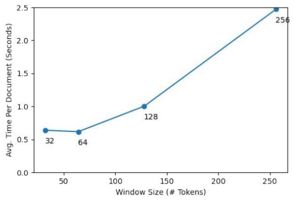


	Mean	Min	Max	STD
Ggzip_Baseline	2.02	1.51	2.61	0.16
Arith_Baseline	2.20	1.64	2.66	0.15
Neural_Varint	4.44	3.22	5.49	0.32
Neural_Arith	4.52	3.05	5.63	0.44

Window Size vs Compression Ratio



Window Size vs Runtime



Discussion

Analysis

- Our results demonstrate the inherent trade-off between compression ratio and compression speed.
- While neural compression approaches are superior in compression ratio, they are inferior in compression speed.
- Smaller windows lead to far better runtime with only negligible reduction in compression ratio
- Compression ratio consistent across file sizes

Future Improvements

- Runtime:
  - Synthesizer attention
  - Fewer layers
  - Improved batching
- Compression ratio
  - Fine-tune with alternative loss term
  - Use 4-bit var-int

References

[1] M. Goyal, K. Tatwawadi, S. Chandak, and I. Ochoa, “DeepZip: Lossless Data Compression using
Recurrent Neural Networks,” ArXiv181108162 Cs Eess Q-Bio, Nov. 2018, Accessed: Feb. 07, 2022.
[Online]. Available: http://arxiv.org/abs/1811.08162

[2] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic Coding for Data Compression,” Commun.
ACM, vol. 30, no. 6, p. 21, 1987.