**Data Compression with Huffman Encoding and its Effects on Energy Consumption**

**Introduction**

**Problem**

There has been a growing concern for *greener* alternatives and supplements in the contemporary technological landscape to address the climate crisis. With the advent of a new technological epoch, artificial intelligence (AI) models such as OpenAI’s ChatGPT or Google’s Gemini have democratized the previously regarded esoteric realm of AI; AI driven software is now easily and widely accessible. On the shoulders of this new age are concerns with the energy costs of this broad technological access. With large language models (LLMs) soon knocking on the door of everyday family households, a new accord for carbon-conscious training software seems material. Almost exclusively, past developments in machine learning (ML) primarily center on optimizing speed or performance, with little to no regard to the impacts on energy consumption (You, Chung, & Chowdhury, 2022). The scale of pop culture’s interest in training personal LLMs may necessitate exploring developments in climate friendly AI software.

**Background**

Interestingly enough, as we were searching for academic research papers on the website [Papers With Code](https://paperswithcode.com/), the top result on the search entry “[Energy Consumption](https://paperswithcode.com/search?q_meta=&q_type=&q=energy+consumption)” has 798 stars. On the same website, the query “[Large Language Models](https://paperswithcode.com/search?q_meta=&q_type=&q=large+language+models)” returns a top result of 122,519 stars. This significant difference in engagement between areas of research -- as shown by the disparity in ratings -- further emphasizes a lack of prioritization on climate-friendly software development within the broader research community.

**Motivation**

A crucial subset of LLM development is training data, and how this data is managed and stored. It’s typically estimated that sophisticated LLMs consume hundreds of terabytes to even petabytes of text data. We want to research the impacts on database size (in bytes) when the textual data is encoded, using Huffman compression algorithms, prior to being stored in a database. The compression algorithms would encode all training data into binary values, then decode when queried – theoretically when training an AI model, alternatively the decoding could be embedded directly into the LLM. Depending on how the sample data is organized or partitioned, each database or database table could have its own Huffman prefix tree specific to that sequester of data. The Huffman algorithms are in Python. We intend to use the Python libraries [pyJoules](https://pypi.org/project/pyJoules/) and [psutil](https://pypi.org/project/psutil/) to survey system resource utilization and energy usage during the encoding and decoding processes to assess any net gains – or losses – in overall energy efficiency. Linux environments also offer the terminal commands *powerstat*, *powertop*, and *turbostat* that provide information about power consumption, CPU usage, and system metrics. The unofficial hypothesis is that a Huffman compressed database will reduce energy overhead, while the cost of the encoding and decoding process may be energy intensive, the overall energy consumed is less than a non-Huffman encoded database.

While these practices and philosophies only represent a small fraction of the broader landscape of LLM development, even modest reductions in carbon emissions can have a significant impact over time. As the scale and magnitude of LLM training continues to grow exponentially, these incremental reductions in carbon emissions can accumulate to create meaningful environmental benefits. Therefore, prioritizing climate-friendly approaches in AI software development, such as the use of Huffman encoding algorithms to reduce energy consumption, is essential for addressing the challenges of the climate crisis in the long term.

# **Works Cited**

Anthony, L. F., Kanding, B., & Selvan, R. (2020, July 6). *Carbontracker: Tracking and Predicting the Carbon Footprint of Training Deep Learning Models.* Retrieved 2024, from paperswithcode: https://paperswithcode.com/paper/carbontracker-tracking-and-predicting-the

You, J., Chung, J.-W., & Chowdhury, M. (2022, August 12). *Zeus: Understanding and Optimizing GPU Energy Consumption of DNN Training.* Retrieved 2024, from arXiv: https://arxiv.org/abs/2208.06102v2