Discover Artificial Intelligence

Research

Green AI: exploring carbon footprints, mitigation strategies, and trade offs in large language model training

Vivian Liu[1] · Yiqiao Yin[1]

Received: 15 December 2023 / Accepted: 26 June 2024
Published online: 08 July 2024

Abstract
Prominent works in the space of Natural Language Processing (NLP) have long attempted to create new innovative models by improving upon previous model training approaches, altering model architecture, and developing more in-depth datasets to better their performance. However, with the quickly advancing field of NLP comes increased greenhouse gas emissions, posing concerns over the environmental damage caused by training LLMs. Gaining a comprehensive understanding of the various costs, particularly those pertaining to environmental aspects, that are associated with artificial intelligence serves as the foundational basis for ensuring safe AI models. Currently, investigations into the CO2 emissions of AI models remain an emerging area of research, and as such, we evaluate the CO2 emissions of well-known large language models, which have an especially high carbon footprint due to their significant amount of model parameters. We argue for the training of LLMs in a way that is responsible and sustainable by suggesting measures for reducing carbon emissions. Furthermore, we discuss how the choice of hardware affects CO2 emissions by contrasting the CO2 emissions during model training for two widely used GPUs. Based on our results, we present the benefits and drawbacks of our proposed solutions and make the argument for the possibility of training more environmentally safe AI models without sacrificing their robustness and performance.

Keywords Natural language processing (NLP) · Greenhouse gas emissions · Environmental damage · Large language models (LLMs) · Artificial intelligence (AI) · Carbon footprint · Hardware choice

1 Introduction

From language translation to smart assistants, the application of large language models (LLMs) in our daily lives has greatly increased. In recent years, the development of LLMs is rapidly expanding, with more and more large corporations hopping on the train of fine-tuning natural language processing (NLP) models. Prior work highlight the ethical and environmental risks of excessively large language models [1]. Moreover, research have advocated for "Green AI" to prioritize sustainability as well as the large carbon footprint of training a 176-billion-parameter LLM [2, 3]. As such, making existing models more efficient and optimizing the performance of models has been a major area of focus [4]. Recent works have looked into improving upon previous methods of training models [5]. For example, a new pretraining approach uses replaced token detection and corrupts the model input by switching the tokens with plausible options sampled from a small generator style network, leading to contextual representations that substantially outperform ones learned by other models given the same model size, data, and compute [6]. Other approaches for improving performance include utilizing new datasets, like WebText, which uses millions of webpages to teach language models natural language processing tasks

✉ Vivian Liu, VivianLiu292@gmail.com; Yiqiao Yin, yy2502@columbia.edu | [1]Columbia University, New York, USA.

without explicit supervision [7]. Research highlighting the approach of generatively pre-training a language model on a varied collection of unlabeled texts, and then applying discriminative fine-tuning, has also demonstrated effectiveness in areas such as commonsense reasoning, question answering, and textual entailment [8]. The research done to make these advancements in the NLP field aim mainly to improve upon NLP models or the way they are trained, but only through a lens of achieving the best performance possible. However, many high-performance NLP models also have a staggeringly large amount of model parameters, and this alludes to an expanding concern for LLMs: how environmentally safe training these models is.

Model training for large language models (such as attention-based NLP models) is highly costly and it opens up a range of ethical issues [9]. As the NLP field expands, leading to increasingly impressive breakthroughs and higher-performing models, so do the costs of the extensive training involved in training these models. The continuously increasing adoption of large language models (LLMs) is driven by their demonstrated versatility and capability in performing various natural language tasks, as highlighted by groundbreaking works such as GPT-3 [4] and the T5 model [10], which showcase their potential for few-shot learning and transfer learning across diverse applications. With our increased reliance on LLMs, the CO2 emissions caused by training NLP models are an issue that absolutely must be discussed in order to drive forward the development of safe AI.

Previous research has attempted to create lighter and more efficient models [3, 11–14]. Some large language models have billions of parameters yet we functionally treat them as black boxes [15]. When using these models, we focus on the results achieved without questioning the various costs of training the models, causing the impact they have on our environment to remain largely undiscussed. In regards to these issues, this paper proposes an experiment using Code Carbon's CO2 emissions tracker to evaluate the CO2 emissions of models as they train and investigate the pros and cons of fine-tuning LLMs [14]. Furthermore, our work assesses the performance of the models with two different data sets and scores them using cosine similarity, a measure of similarity between vectors, and semantic textual similarity (STS), a measure of semantic equivalence between two texts [16, 17].

Aside from carbon emissions, other considerations we account for include the financial implications of implementing each strategy to lower emissions. Challenges with utilizing transfer learning also arise in effectively deploying LLMs under resource limitations, particularly when operating within constrained computational training and inference budgets [12]. As such, our objective is to discuss the various aspects of these possible solutions to evaluate the feasibility of applying them to lower CO2 emissions. Moreover, we seek to ascertain whether our proposed strategies for mitigating environmental damage are truly accessible to the layperson aspiring to train LLMs.

This research aims to explore how the training of large language models (LLMs) in Natural Language Processing (NLP) can be made more environmentally sustainable without compromising their robustness and performance. To achieve this, we evaluate the CO2 emissions of well-known LLMs, particularly noting the significant emissions due to their extensive parameters. We propose strategies to reduce carbon emissions and examine the impact of different hardware choices, specifically comparing two widely used GPUs, on CO2 emissions during model training. Our findings highlight the advantages and disadvantages of these approaches, demonstrating the potential for training environmentally safe AI models without losing their effectiveness. Through this investigation, we seek to contribute to the emerging research on the environmental impact of AI models and advocate for responsible and sustainable practices in the development and training of LLMs. In addition, this research addresses the growing concern about the environmental impact of training large language models (LLMs) with extensive parameters by assessing their CO2 emissions, suggesting emission reduction strategies, and examining how different hardware choices affect their environmental footprint. Our objective is to discuss these aspects and evaluate the feasibility of applying these solutions to lower CO2 emissions, demonstrating that it is possible to train high-performance, environmentally sustainable AI models without sacrificing their effectiveness.

Based on our experiment, we are able to present the analysis of different strategies to lower CO2 emissions. The results from the training and testing of LLMs lead us to propose that balancing impeccably robust and high-performing models with strategies that commendably reduce CO2 emissions to ensure a sustainable future is not just a mere possibility, but a tangible reality within our reach.

2 Related work

Previous research has attempted to make models lighter and more efficient. For example, Albert is a lighter version of the well-known BERT model with reduced parameters and faster training times [11]. Moreover, DistilBERT similarly reduces the parameters of the BERT model, lowering model parameters by 40% [12]. Using these lighter models may help us reduce the carbon footprint of model training, and as such these papers are undeniably taking a step in the right direction by creating models that are lighter and more affordable to train. However, much of this research focuses on lowering the monetary cost of training and for the most part neglects to discuss the importance of decreasing CO2 emissions.

Aside from that, there has been research done to create carbon emissions trackers and calculators that can approximate the CO2 emissions of models [13]. This has led to the release of open-source packages intended to help data scientists and researchers track the carbon emissions of their language models [14]. Other recent works have further attempted to start shedding light on the importance of lowering the carbon emissions produced in model training by surveying factors that may influence the CO2 emissions produced by machine learning models [3].

2.1 Motivation

Apprehensions arise from the rapid advancements witnessed in the NLP domain. In the absence of timely attention to environmental concerns, there exists a legitimate concern that this unchecked progress may result in irreversible environmental harm in the foreseeable future. Thus, our goal is to generate usable metrics quantifying model performance and CO2 emissions which will then enable corporations to utilize large language models to make informed and sustainable decisions when training NLP models. We also attempt to analyze different approaches for reducing CO2 emissions to help individuals take actionable efforts to lower the CO2 emissions when training their NLP models.

We additionally attempt to present an accurate picture of the pricing of possible solutions for lowering CO2 emissions. By analyzing the financial costs of implementing certain strategies to lower CO2 emissions, we can answer the question, “Is training AI sustainably truly something that anybody can do, or is it only a hope for large corporations which have better access to more novel solutions for increasing CO2 emission efficiency?”

3 Experimental setup

To formulate strategies for reducing the CO2 emissions of model training and determine whether they are financially accessible to any person wanting to train LLMs, an initial prerequisite involves the selection of suitable models for performance and emissions assessment. Furthermore, the selection of appropriate datasets for model training and subsequent evaluation, as well as the designation of a set of quantifiable metrics to gauge model proficiency, are essential determinations in this process.

The principal stages of the experiment are depicted in Fig. 1. This experiment involves the fine-tuning of a variety of large language models, including BERT, DistilBERT, and T5, using different tokenizers and the SQuAD dataset [18]. Throughout the fine-tuning phase, parameters such as loss, training time, and CO2 emissions are meticulously documented. Subsequently, the performance of the trained models is assessed using an out-of-sample test set derived from both the SQuAD and AdversarialQA datasets. Evaluation metrics employed for this purpose include cosine similarity and Semantic Textual Similarity (STS) scores.

3.1 Models

Transformer models, when enhanced through self-supervised pretraining, have dramatically transformed the fields of natural language processing (NLP) and information retrieval (IR) [19]. They yield superior outcomes across multiple domains and tasks. As a result, we chose three renowned transformers for performance evaluation: BERT, DistilBERT, and T5.

BERT, an acronym for Bidirectional Encoder Representations from Transformers, is a model that has been pre-trained using Masked Language Modeling and Next Sentence Prediction techniques, making it highly effective for a variety of NLP tasks. The innovation behind BERT lies in its approach to pre-train comprehensive bidirectional representations from unlabelled text by simultaneously considering both left and right context across all layers. Consequently, the pre-trained BERT model can be easily fine-tuned by adding just a single additional output layer, enabling the creation of cutting-edge models for a broad spectrum of tasks, such as question answering and language inference, without the need for significant task-specific modifications to the architecture. BERT has achieved unprecedented state-of-the-art results in eleven natural language processing tasks, including notable enhancements over prior models in metrics such as GLUE

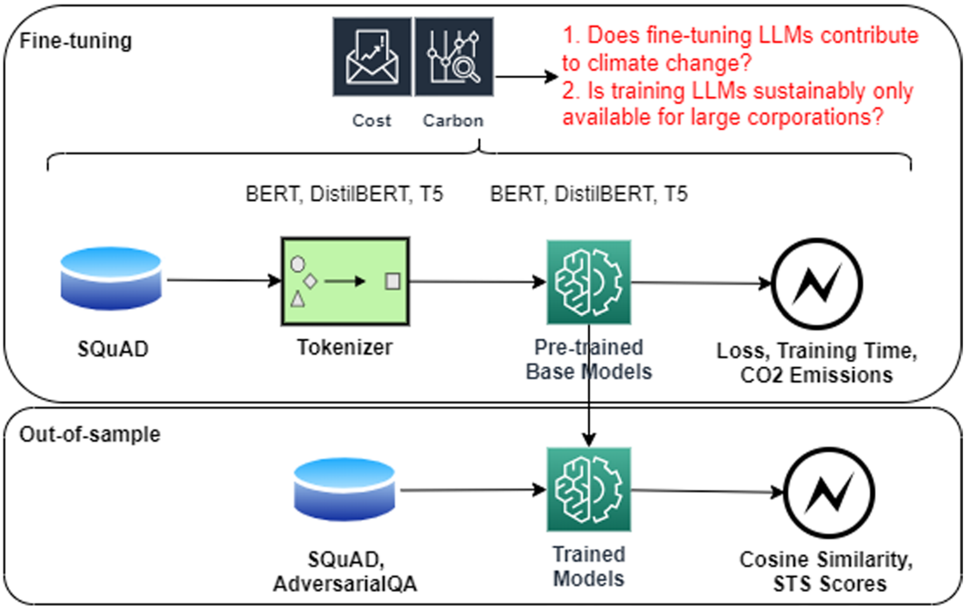
Fine-tuning
Cost
Carbon
1. Does fine-tuning LLMs contribute to climate change?
2. Is training LLMs sustainably only available for large corporations?
BERT, DistilBERT, T5
BERT, DistilBERT, T5
SQuAD
Tokenizer
Pre-trained Base Models
Loss, Training Time, CO2 Emissions
Out-of-sample
SQuAD, AdversarialQA
Trained Models
Cosine Similarity, STS Scores

Fig. 1 Flowchart of the process used for training, testing, and recording measurements of model performance

score, MultiNLI accuracy, SQUAD v1.1 question answering Test F1, and SQUAD v2.0 Test F1 [20]. The attention mechanisms within BERT demonstrate patterns like focusing on delimiter tokens, certain positional offsets, or covering entire sentences more broadly. Some attention heads align closely with linguistic concepts of syntax and co-reference [21].

DistilBERT, in contrast, is a streamlined iteration of the original BERT model, utilizing knowledge distillation techniques to dramatically decrease the model's parameter count, thus yielding a more compact version. This process involves pre-training a smaller, general-purpose language representation model, known as DistilBERT, which can subsequently be fine-tuned to achieve commendable performance across a diverse array of tasks, akin to its more substantial predecessors. Through the application of knowledge distillation during its pre-training phase, DistilBERT manages to reduce the BERT model's size by 40%, while preserving 97% of its capacity to understand language and achieving a speed increase of 60%. The introduction of a triple loss, which combines language modeling, distillation, and cosine-distance losses, capitalizes on the larger models' pre-training inductive biases. This results in a model that is not only smaller, faster, and lighter but also more cost-effective to pre-train [12].

T5, the fifth generation of a transformer model, is capable of performing tasks presented to it in a text-to-text format [22]. To further explore the models' performance, we also conducted experiments with different tokenizers: bert-base-cased, distilbert-base-uncased, and t5-base tokenizers. These tokenizer variations allowed us to investigate how tokenization strategies might impact the models' effectiveness in handling text data and CO2 emissions.

Beyond the base models themselves, we also tried using BERT's tokenizer with a DistilBERT base model and we combined DistilBERT's tokenizer with a BERT base model. This was done to develop an understanding of how the tokenizer selected affects a model's CO2 emissions and model performance.

3.2 Data

The pre-trained models selected were trained using the Stanford Question Answering Dataset (SQUAD), a dataset comprising of over 100,000 questions created by crowdworkers that is based on Wikipedia articles [18]. Each question from the dataset requires an answer that can be found within the corresponding reading passage in order to test reading comprehension. The dataset is analyzed to determine the reasoning types needed to answer the questions, with a particular focus on dependency and constituency trees. During testing, it was observed that human participants significantly surpassed the performance of both a sophisticated logistic regression model and a baseline model on the dataset. This suggests that the dataset presents a formidable challenge for upcoming studies in the domain of reading comprehension. [18].

Before training the models, the data was split into training and validation, with a training-validation split of 20% used for model training. This meant that of the 107,785 question-answer pairs contained in the SQUAD dataset, 21,557

Table 1 Model performance trained on SQUAD data, 3 epochs, and T4 GPU. Model consists of BERT (named as B.), DistilBERT (named as D.), and T5

Model	Tokenizer	Loss	Time	CO2 (kg)	RAM (W)	Model Param.
B.	D.	1.5	1152	1.15E-02	4.8	1.1 bn
B.	B.	3.0	1183	1.18E-02	9.5	1.1 bn
D.	D.	1.4	634	6.28E-03	9.5	66 mm
D.	B.	3.3	646	1.12E-02	9.5	66 mm
t5	t5-base	1.0	22996	8.07E-02	9.5	11 bn

Tokenizer consists of bert-base-cased (named as B.), distilbert-base-uncased (named as D.), and t5-base. Loss is the validation loss and time is the time taken in seconds for training. CO2 is CO2 emissions and RAM is RAM usage from training. Model Param. is the model parameters of each model. The name "bn" and "mm" represent billion and million, respectively

Table 2 Model performance trained on SQUAD data, 3 epochs, and A100 GPU. Model consists of BERT (named as B.), DistilBERT (named as D.), and T5

Model	Tokenizer	Loss	Time	CO2 (kg)	RAM (W)	Param.
B.	D.	1.4	221	3.20E-05	31.3	1.1 bn
B.	B.	3.1	233	4.40E-03	31.3	1.1 bn
D.	D.	1.5	142	1.60E-05	31.3	66 mm
D.	B.	3.2	598	5.30E-03	9.5	66 mm
t5	t5-base	1.0	7661	3.70E-04	31.3	11 bn

Tokenizer consists of bert-base-cased (named as B.), distilbert-base-uncased (named as D.), and t5-base. Loss is the validation loss and time is the time taken in seconds for training. CO2 is CO2 emissions and RAM is RAM usage from training. Param. is the model parameters of each model. The name "bn" and "mm" represent billion and million, respectively

question-answer pairs were used for evaluating the models' performance between epochs, while the remaining 86,228 question-answer pairs were used purely for the actual training of each model.

Furthermore, we utilized the AdversarialQA dataset as a held-out test set to gauge the performance of our trained NLP models on data to which they had not been previously exposed. The AdversarialQA dataset was conceived through an adversarial human annotation process, which incorporated both a human annotator and an NLP model in the creation of question-answer pairs. Designed to scrutinize a model's reading comprehension capabilities, this dataset comprises approximately 36,000 question-answer pairs [23].

3.3 Metrics

As previously mentioned, to gain a comprehensive understanding of the performance of various models and to compare their performance with their corresponding carbon emissions, we employed the emissions tracker from the CodeCarbon Python package to provide us with an estimate for the amount of CO2 emissions generated during model training. From the training of the various models, we recorded essential metrics such as loss, training time, CO2 emissions, RAM usage, and model parameters (presented in Table 1 and Table 2).

The first table provides an overview of the models' performance when trained using the T4 GPU (see detailed results in Table 1). It includes the validation loss measured for each model and the corresponding estimated amount of CO2 emissions in kilograms generated during the training process.

In contrast, the second table showcases the results of training the models with the A100 40GB SXM GPU (see detailed results in Table 2). It also presents the validation loss and the estimated CO2 emissions produced during the training sessions. By juxtaposing these two tables, we can gain valuable insights into the trade-offs between model performance and environmental impact based on the type of GPU utilized during the training process.

In the third table, we conducted 100 random samplings from the SQUAD dataset to evaluate the performance of the different models (see detailed results in Table 3). Each model was tasked with making inferences using 100 queries from the dataset, and we measured their performance using cosine similarity, STS (BERT-Embedding), STS (OpenAI-Embedding), and STS (Palm-Embedding). The formula for cosine similarity is defined in the following equation.

Table 3 Model performance scored on 100 observations sampled randomly from SQUAD data. Model consists of BERT (named as B.), DistilBERT (named as D.), and T5

Model	Token.	Param.	Cosine	BERT	OpenAI	Palm
B.	D.	1.1 bn	0.27 (0.39)	0.38 (0.38)	0.82 (0.10)	0.67 (0.17)
B.	B.	1.1 bn	0.03 (0.14)	0.20 (0.19)	0.78 (0.05)	0.59 (0.09)
D.	D.	66 mm	0.20 (0.30)	0.34 (0.31)	0.81 (0.08)	0.65 (0.14)
t5	t5-base	11 bn	0.23 (0.26)	0.54 (0.25)	0.88 (0.06)	0.77 (0.10)

Tokenizer (named as token.) consists of bert-base-cased (named as B.), distilbert-base-uncased (named as D.), and t5-base. Param. is the model parameters of each model. Cosine is the cosine similarity, BERT is the STS (BERT) score, OpenAI is the STS (OpenAI) score, and Palm is the STS (Palm) score. The name "bn" and "mm" refer to billion and million, respectively

Table 4 Model performance scored on 100 observations sampled randomly from AdversarialQA(held-out test set) data

Model	Token.	Param.	Cosine	BERT	OpenAI	Palm
B.	D.	1.1 bn	0.11 (0.23)	0.28 (0.26)	0.80 (0.06)	0.62 (0.11)
B.	B.	1.1 bn	0.04 (0.12)	0.20 (0.18)	0.77 (0.04)	0.59 (0.09)
D.	D.	66 mm	0.09 (0.22)	0.24 (0.25)	0.78 (0.06)	0.60 (0.11)
t5	t5-base	11 bn	0.06 (0.14)	0.23 (0.23)	0.80 (0.05)	0.63 (0.10)

Model consists of BERT (named as B.), DistilBERT (named as D.), and T5. Tokenizer (named as token.) consists of bert-base-cased (named as B.), distilbert-base-uncased (named as D.), and t5-base. Param. is the model parameters of each model. Cosine is the cosine similarity, BERT is the STS (BERT) score, OpenAI is the STS (OpenAI) score, and Palm is the STS (Palm) score. The name "bn" and "mm" refer to billion and million, respectively

$$similarity = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}|| \times ||\mathbf{B}||} \tag{1}$$

The values presented in the table represent the average performance scores, and the standard deviation is listed in parentheses. These results were obtained by calculating the models' performance over 100 samples, providing a comprehensive assessment of their overall effectiveness and consistency in handling the dataset queries.

In the fourth table, we continue the evaluation of model performance over 100 samples (see detailed results in Table 4). However, in this case, the models were assessed using the AdversarialQA dataset, which serves as a held-out test set, rather than the SQUAD dataset. Each model was subjected to the same inference process, and their performance was measured using cosine similarity, STS (BERT-Embedding), STS (OpenAI-Embedding), and STS (Palm-Embedding) metrics. Similar to the third table, the values in the fourth table represent the average performance scores, with the standard deviation indicated in parentheses. This evaluation using the AdversarialQA dataset allows us to further validate the models' capabilities in handling different types of queries and assess their robustness across diverse datasets.

We used the average carbon intensity value provided by Code Carbon's emissions tracker, which is consistent across all benchmarks in our study. This ensures that the CO2 emissions estimates are based on a uniform metric for all the models and GPUs evaluated.

Runtime/resource usage refers to the time and computational resources (CPU and GPU usage) consumed during model training, including the duration of the training process and the hardware utilized. Electricity usage is the amount of electrical energy consumed by the hardware during training, measured in kilowatt-hours (kWh), and depends on the power draw of the CPU, GPU, and RAM. Carbon intensity, measured in kg CO2 per kWh, represents the CO2 emissions per unit of electricity consumed and varies by region based on the local energy mix. These distinctions help clarify where variability in our results arises. Additionally, we have specified that the emissions reported are specifically CO2 emissions, as inferred from electricity usage using the stated carbon intensity value, and we have labeled them accordingly throughout the paper. If the emissions included all greenhouse gases, CO2e would be a more appropriate notation.

4 Results and discussion

We employ data from model training, testing, and the CO2 emissions tracker to evaluate the advantages and disadvantages of specific models and GPUs. By examining the merits of certain models, we can draw conclusions regarding which techniques are efficient in reducing CO2 emissions.

4.1 Climate and environmental protection

In our study, we have focused specifically on the environmental impact of CO2 emissions resulting from model training. While we acknowledge that CO2 emissions are just one aspect of environmental sustainability, they are a significant and quantifiable metric that can provide valuable insights into the energy efficiency of different models and hardware configurations. By using Code Carbon's emissions tracker, we aim to provide a clear and consistent measure of CO2 output, which serves as an indicator of the environmental cost associated with AI model training. We recognize that a comprehensive environmental assessment would include other factors such as water usage, electronic waste, and more; however, the scope of this study is intentionally limited to CO2 emissions to allow for focused and detailed analysis. We have highlighted this limitation and the rationale behind it in the introduction to ensure clarity and context for our readers.

Within the scope of our experiment, we discuss the environmental consequences of model training in the form of CO2 emissions and use the CO2 emissions estimates provided by Code Carbon's emissions tracker to quantify how sustainable certain models and GPU options are.

The experimental results reveal that both the T5 and BERT models emitted considerably more CO2 compared to DistilBERT (see Tables 1, 2). Similarly, utilizing a bert-base-cased tokenizer also resulted in higher CO2 emissions compared to using a distilbert-base-uncased tokenizer. During model training with the T4 GPU, the DistilBERT model with a distilbert-base-uncased tokenizer had 46.9% fewer CO2 emissions in comparison to the BERT model with a tokenizer of bert-base-cased.

Contrasting tables1 and 2 clearly demonstrates the substantial reduction in model training time and CO2 emissions when switching from the T4 GPU to the A100 GPU. On average the A100 GPU decreased model training time by 62.6%. It produced especially large reductions in training time for the BERT model with a tokenizer of distilbert-base-uncased and the BERT model with a tokenizer of bert-base-cased which experienced 80.8% and 80.3% reductions in training time respectively.

The A100 GPU also lowered carbon emissions by a staggering 83% on average across all five models. It remarkably reduced CO2 emissions by 99.7% for both BERT with a tokenizer of distilbert-base-uncased and DistilBERT with a tokenizer of distilbert-base-uncased, 67.2% for BERT with a tokenizer of bert-base-cased, 53.1% for DistilBERT with a tokenizer of bert-base-cased, and 99.5% for the T4 model.

While CO2 emissions are only one aspect of environmental sustainability, they serve as a significant and quantifiable metric that provides valuable insights into the energy efficiency of different models and hardware configurations. By using Code Carbon's emissions tracker, we aim to offer a clear and consistent measure of CO2 output, highlighting the environmental cost associated with AI model training.

Our paper contextualizes the investigation of CO2 emissions by comparing the environmental impact of training well-known large language models (LLMs) such as BERT, DistilBERT, and T5 using different hardware configurations (T4 and A100 GPUs). We utilize Code Carbon's emissions tracker to provide consistent and quantifiable measurements of CO2 output, highlighting the significant carbon footprint associated with models having extensive parameters. This approach allows us to benchmark the CO2 emissions across various models and hardware setups, offering insights into more sustainable practices in AI model training.

In our work, we used "CodeCarbon", a Python package that estimates the running hardware electricity power consumption (GPU + CPU + RAM) and the library applies the results to the carbon intensity of the region where the computing is done [24].

From "CodeCarbon" library, training of a neural network model can be measured. The results provide both Energy consumed for RAM (measured in kWh) and RAM Power (measured in W). The energy consumption for RAM (measured in kWh) indicates the total amount of electrical energy consumed by the RAM during the neural network model training process. For example, training a two-layer neural network model produced 0.000020 kilowatt-hours (kWh),

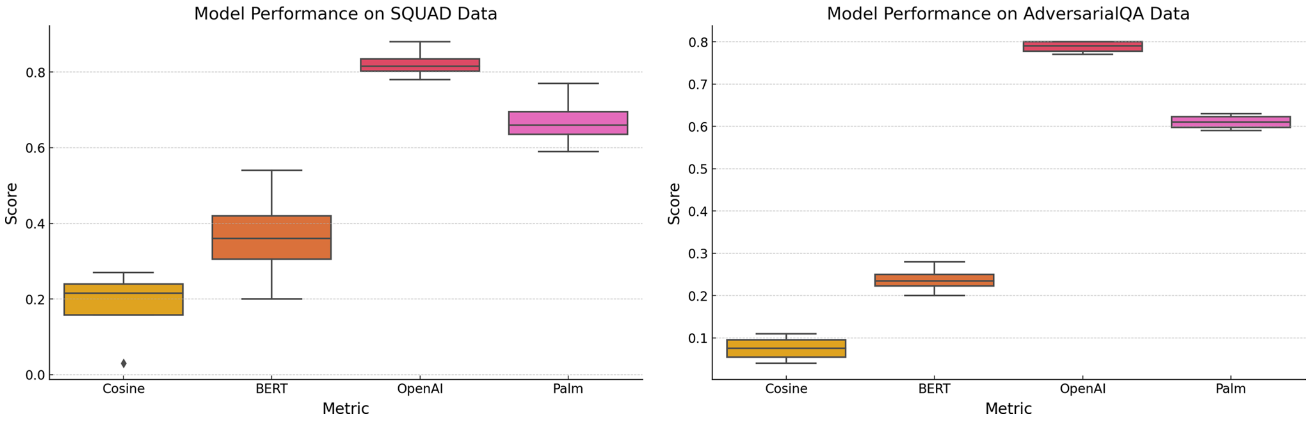

Fig. 2 Confidence comparison. The figure presents boxplots to compare the variability of the candidates discussed in the paper

which is a very small amount of energy. One kilowatt-hour is equivalent to using 1000 watts of power for one hour. Therefore, this value represents the energy usage of the RAM over the duration of the training. The RAM power (measured in W) represents the power consumption of the RAM, measured in watts (W). Power is the rate at which energy is consumed. In this toy example of training a two-layer neural network model, the RAM consumed approximately 4.75 watts of power during the training process. This value is an average power consumption over the time period that the model was being trained.

4.2 Performance

However, despite being more lightweight compared to its BERT counterpart, models using a distilbert-base-uncased tokenizer demonstrated better performance than models using a bert-base-cased tokenizer on both the SQUAD and AdversarialQA datasets. In fact, models with a distilbert-base-uncased tokenizer exhibited lower loss and higher cosine similarity, STS (BERT-Embedding), STS (OpenAI-Embedding), and STS (Palm-Embedding) scores than models with the bert-base-cased tokenizer. The distilbert-base-uncased tokenizer models' consistent higher performance over many scoring metrics demonstrates that it is robust as well as high-performance. Beyond decreasing carbon emissions, the DistilBERT model with a distilbert-base-uncased tokenizer lowered the time taken to train by 46% and decreased loss by 54.5% compared to the BERT model with bert-base-cased as its tokenizer. This model produced similar reductions in training time, carbon emissions, and loss compared to BERT with a bert-base-cased tokenizer when trained while using the A100 GPU. The BERT model with a tokenizer of distilbert-base-uncased also had lower loss than the BERT model with a bert-base-cased tokenizer, experiencing a 51.3% decrease in loss. These consistent findings underscore that the distilbert-base-uncased tokenizer did not sacrifice any of the efficacy of the bert-base-cased tokenizer, as it consistently outperforms models employing the bert-base-cased tokenizer across various evaluation metrics and datasets.

By training the T5 model on the SQUAD dataset for three epochs, it was able to perform better in validation loss than the other models trained on three epochs. However, in Table 3 and Table 4, the T5 model trained for only one epoch does not perform the best in all scoring categories. The experimental results reveal the general correlation between the number of epochs a model is trained for and its performance. Unfortunately, training for more epochs subsequently leads to higher CO2 emissions.

Despite its CO2 emissions with the T4 GPU, the T5 model trained on the A100 GPU emitted less CO2 than models utilizing a BERT tokenizer and demonstrated a significantly better performance (see Table 2). It outperformed BERT with a bert-base-cased tokenizer with a reduction in loss of 66.8% and BERT with a distilbert-base-uncased tokenizer with a reduction in loss of 68%.

To visualize the confidence intervals, we present the information in the following boxplots (Fig. 2).

Another remarkable result from our model training and testing was that using the A100 GPU had negligible impact on model performance, and in certain cases, models trained with the A100 GPU even outperformed those trained with the T4 GPU. However, for the majority of cases, the loss values for each model remained relatively consistent across both GPUs. The consistent performance of models between both GPUs when considered in combination with the lowered training time and carbon emissions of models trained on the A100 GPU evidences the possibility of lowering CO2 emissions without sacrificing model performance in LLM training.

The experimental results show that the A100 GPU significantly reduces both model training time and CO2 emissions compared to the T4 GPU. On average, the A100 GPU decreased model training time by 62.6% and lowered carbon emissions by 83% across all models tested. For specific models, such as BERT with distilbert-base-uncased and BERT with bert-base-cased tokenizers, the A100 GPU reduced training time by over 80% and CO2 emissions by up to 99.7%. The T5 model, when trained on the A100 GPU, emitted less CO2 and demonstrated significantly better performance compared to models utilizing a BERT tokenizer. It outperformed BERT with a bert-base-cased tokenizer with a 66.8% reduction in loss and with a distilbert-base-uncased tokenizer with a 68% reduction in loss. Despite its higher CO2 emissions with the T4 GPU, the T5 model showed improved efficiency when switched to the A100 GPU. The A100 GPU generally outperforms the T4 GPU in terms of reducing training time and CO2 emissions, making it a more efficient and environmentally friendly option for model training. However, it is not a one-size-fits-all solution. The effectiveness of a GPU can vary depending on the specific model and configuration being used. For instance, the T5 model benefits significantly from the A100 GPU, but other models like BERT also show substantial improvements.

The T5 model experienced substantial reductions in training time when using the A100 GPU. This decrease in training time can be highly relevant in practice, as it allows for quicker iterations and faster development cycles. For organizations and researchers working with large-scale models, these time savings can translate into increased productivity and shorter time-to-market for new applications. The A100 GPU dramatically reduced the CO2 emissions associated with training the T5 model. In the context of growing concerns about the environmental impact of AI, these reductions are significant. Organizations aiming to reduce their carbon footprint will find the gains relevant, as they help align with sustainability goals and regulatory requirements. The T5 model showed improved performance metrics when trained on the A100 GPU compared to the T4 GPU. This improvement in validation loss and other performance metrics means that the models trained on the A100 GPU are not only more efficient but also potentially more effective in their application domains. Higher performance can lead to better results in tasks such as natural language processing, which is highly relevant for practical applications. While the A100 GPU is more expensive than the T4 GPU, the gains in training time, CO2 emissions, and performance can justify the investment, especially for organizations with high computational demands. The cost savings from reduced energy consumption and the value derived from faster and more accurate models can offset the higher initial expenditure. As such, the gains achieved by using the A100 GPU for training the T5 model are indeed relevant in practice. The significant improvements in training time, CO2 emissions, and model performance make it a valuable option for organizations and researchers dealing with large-scale models and looking to optimize their workflows and environmental impact.

4.3 Discussion of CO2 mitigation strategies

Although it may seem that based on the T5 model's low validation loss and high CO2 emissions, it is unrealistic to have a model be both optimized for performance and low CO2, this does not mean the ideal of achieving high performance while being environmentally conscientious is impossible. Based on the significant improvements on BERT brought by its distilled counterpart, DistilBERT, there may be potential to decrease CO2 emissions in the future through efforts to reduce the model parameters of existing models, thereby capitalizing on the performance benefits of high-performance models that currently produce significant CO2 emissions while simultaneously allowing for the mitigation of their environmental consequences.

This could be applied to various other models as well, including some of the other prominent LLMs that were not mentioned in the conducted training and testing of models in this paper. Transformer-XL, as an illustration, facilitates the learning of dependencies that exceed a predetermined length while maintaining temporal coherence intact. It delivers enhanced results across both brief and extended sequences and boasts an evaluation speed that is over 1,800 times quicker than that of standard Transformers [25]. Albert, another model, employs a pair of parameter-reduction strategies to decrease memory usage and accelerate the training process of BERT [11]. These models may fit certain niches better than the models we tested and be fine-tuned for optimization on specific tasks. This is why we propose the implementation of certain strategies like lightening existing models and using faster GPUs that are generally applicable so that regardless of the model being used, corporations can stay environmentally conscious and have methods available for reducing the CO2 emissions of training their models.

Additionally, the results achieved with the A100 GPU highlight the significant benefits of employing faster hardware for environmental sustainability and lowering training time; The A100 GPU proved to significantly increase CO2 emission efficiency, all without compromising the models' overall performance. Despite this, it is important to acknowledge a notable downside associated with utilizing an A100 GPU in contrast to a T4 GPU.

We acknowledge the importance of incorporating a comprehensive evaluation framework that includes sequence labeling, which is fundamental in NLP tasks such as Part-of-Speech (POS) tagging, Named Entity Recognition (NER), chunking, semantic role labeling, and text segmentation. However, the primary focus of our current work is on assessing and comparing CO2 emissions under different experimental setups. While we recognize the value of a more extensive evaluation, including sequence labeling, our aim here is to highlight the environmental impact of training large language models. We will certainly consider including a broader evaluation framework in future work to provide a more holistic assessment of model performance across various NLP tasks.

4.4 Discussion of financial implication

One prominent drawback of utilizing the A100 GPU for training is that the net cost of using it is higher. In our experiment, the A100 GPU increased RAM usage, which can result in higher expenses for companies and individuals utilizing it for model training. Additionally, the base price to buy the GPU is also more expensive, e.g. one listing on Amazon sells the GPU for $6,799.[1] This is comparatively much more expensive than the T4 GPU sold on the same website for a listing of $1,175.99.[2] This factor poses a concern, as the cost implications might limit its feasibility as an accessible option for many. This begs the question of whether training large language models in an environmentally sustainable way is truly an affordable option for any individual looking to conduct NLP model training. The trade-off between its environmental benefits and the associated financial implications necessitates careful consideration and further exploration of cost-effective alternatives to make sustainable AI practices more widely attainable.

With that said, for individuals without the budget to use a more expensive, faster GPU to decrease emissions, there are still other options. In fact, using a lighter model or modifying an existing model to have fewer model parameters are both viable strategies for saving on computational costs, as well as providing the added benefit of lowering CO2 emissions.

We want to clarify that our paper does not assume only large companies can consider environmental impacts. We understand how readers might infer this conclusion, but our intention was to highlight that large companies often have the resources to invest in a higher volume of GPUs, which naturally leads to more significant computational capacity and, consequently, a higher environmental impact. We do not imply that smaller entities are incapable of contributing to environmentally sustainable practices. Instead, we emphasize that larger companies, having already invested heavily in GPU infrastructure, should be particularly mindful of their environmental footprint. Our discussion aims to urge those with extensive computational resources to consider the environmental implications of their operations. Nonetheless, we also provide strategies for individuals and smaller entities to reduce computational costs and emissions, such as using lighter models or reducing model parameters, thereby promoting sustainable AI practices across all scales.

5 Conclusion

By implementing straightforward yet effective strategies, such as employing lighter models and faster GPUs, we can significantly lower a model's CO2 emissions without compromising model robustness and performance. To do so comes with certain drawbacks, such as allocating an appropriate budget for better GPUs. It additionally requires the selection of lighter models or at least the expenditure of effort into reducing model parameters of existing models. However, this is a necessary step forward for the NLP field as embracing environmentally safe practices while maintaining results will not only be necessary to reduce the environmental damage of large language model training, but also promises new horizons for sustainable and robust, high-performing models in the NLP field through added benefits in model efficiency and speed.

Author contributions V.L. wrote the main manuscript text. V.L. conducted the experiments and recorded the figures. Y.Y. proofread the paper and agreed with the results. All authors reviewed the manuscript.

[1] For the listing of the A100 GPU, please see https://www.amazon.com/NVIDIA-Tesla-A100-Ampere-Graphics/dp/B0BGZJ27SL

[2] For the listing of the T4 GPU, please see https://www.amazon.com/PNY-Datacenter-Express-Passive-Cooling/dp/B07QF9MJFR.

Data availability The data sets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Code availability Code for data cleaning and analysis is provided upon requests as part of the proposed statistical package. It will be accessible by request once the paper has been conditionally accepted.

Declarations

Competing interests The authors declare no competing interests.

References

1. Bender EM, Gebru T, McMillan-Major A, Shmitchell S. On the dangers of stochastic parrots: Can language models be too big? In Proceedings of the 2021 ACM conference on fairness, accountability, and transparency, 2021:610–623.
2. Schwartz R, Dodge J. Noah a Smith, and Oren Etzioni. Green ai Commun ACM. 2020;63(12):54–63.
3. Luccioni AS, Hernandez-Garcia A. Counting carbon: a survey of factors influencing the emissions of machine learning. *arXiv preprint*arXiv:2302.08476*, 2023.*
4. Brown T, Mann B, Ryder N, Subbiah M, Kaplan JD, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, et al. Language models are few-shot learners. Adv Neural Inf Process Syst. 2020;33:1877–901.
5. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint*arXiv:1907.11692*, 2019.*
6. Clark K, Luong MT, Le QV, Manning CD. Electra: pre-training text encoders as discriminators rather than generators. *arXiv preprint*arXiv:2003.10555*, 2020.*
7. Radford A, Jeffrey W, Child R, Luan D, Amodei D, Sutskever I, et al. Language models are unsupervised multitask learners. OpenAI blog. 2019;1(8):9.
8. Radford A, Narasimhan K, Salimans T, Sutskever I, et al. Improving language understanding by generative pre-training. OpenAI, 2018.
9. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. Adv Neural Inf Process Syst, 2017;30.
10. Colin R. Exploring the limits of transfer learning with a unified text-to-text transformer. JMLR. 2020;21(140):1.
11. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Soricut R. Albert: a lite bert for self-supervised learning of language representations. *arXiv preprint*arXiv:1909.11942*, 2019.*
12. Sanh V, Debut L, Chaumond J, Wolf T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint*arXiv:1910.01108*, 2019.*
13. Lacoste A, Luccioni A, Schmidt V, Dandres T. Quantifying the carbon emissions of machine learning. *arXiv preprint*arXiv:1910.09700*, 2019.*
14. Budennyy SA, Lazarev VD, Zakharenko NN, Korovin AN, Plosskaya OA, Dimitrov DV, Akhripkin VS, Pavlov IV, Oseledets IV, Barsola IS, Egorov IV, et al. Eco2ai: carbon emissions tracking of machine learning models as the first step towards sustainable ai. In Doklady Mathematics, volume 106, pages S118–S128. Springer, 2022.
15. Hanyao Huang O, Zheng DW, Yin J, Wang Z, Ding S, Yin H, Chuan X, Yang R, Zheng Q, et al. Chatgpt for shaping the future of dentistry: the potential of multi-modal large language model. Int J Oral Sci. 2023;15(1):29.
16. Xia Peipei, Zhang Li, Li Fanzhang. Learning similarity with cosine similarity ensemble. Inf Sci. 2015;307:39–52.
17. Agirre E, Bos J, Diab M, Manandhar S, Marton Y, Yuret D. Semeval-2012 task 6: a pilot on semantic textual similarity. In * SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012), 2012:385–393.
18. Rajpurkar P, Zhang J, Lopyrev K, Liang P. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras, editors, Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.
19. Lin Jimmy, Nogueira Rodrigo, Yates Andrew. Pretrained transformers for text ranking: Bert and beyond. Berlin: Springer Nature; 2022.
20. Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint*arXiv:1810.04805*, 2018.*
21. Clark K, Khandelwal U, Levy O, Manning CD. What does bert look at? an analysis of bert's attention. *arXiv preprint*arXiv:1906.04341*, 2019.*
22. Bird JJ, Ekárt A, Faria DR. Chatbot interaction with artificial intelligence: human data augmentation with t5 and language transformer ensemble for text classification. J Amb Intell Human Comput. 2023;14(4):3129–44.
23. Bartolo M, Roberts A, Welbl J, Riedel S, Stenetorp P. Beat the ai: investigating adversarial human annotation for reading comprehension. Trans Assoc Comput Linguistics. 2020;8:662–78.

24. Courty B, Schmidt V, Luccioni S, Goyal-Kamal, Coutarel M, Feld B, Lecourt J, Connell J, Saboni A, Inimaz, supatomic, Mathilde Léval, Blanche L, Cruveiller A, ouminasara, Zhao F, Joshi A, Bogroff A, de Lavoreille H, Laskaris N, Abati E, Blank D, Wang Z, ArminCatovic, Marc Alencon, Michał Stęchły, Bauer C, Otavio L, JPW, and MinervaBooks. mlco2/codecarbon: v2.4.1, May 2024.
25. Dai Z, Yang Z, Yang Y, Carbonell J, Le QV, Salakhutdinov R. Transformer-xl: attentive language models beyond a fixed-length context. *arXiv preprint*arXiv:1901.02860*, 2019.*