DOI:10.1145/3381831

Creating efficiency in AI research will decrease its carbon footprint and increase its inclusivity as deep learning study should not require the deepest pockets.

BY ROY SCHWARTZ, JESSE DODGE, NOAH A. SMITH, AND OREN ETZIONI

Green AI

SINCE 2012, THE field of artificial intelligence (AI) has reported remarkable progress on a broad range of capabilities including object recognition, game playing, speech recognition, and machine translation.[43] Much of this progress has been achieved by increasingly large and computationally intensive deep learning models.[a] Figure 1, reproduced from Amodei et al.,[2] plots training cost increase over time for state-of-the-art deep learning models starting with AlexNet in 2012[24] to AlphaZero in 2017.[45] The chart shows an overall increase of 300,000x, with training cost doubling every few months. An even sharper trend can be observed in NLP word-embedding approaches by looking at ELMo[34] followed by BERT,[8] openGPT-2,[35] XLNet,[56] Megatron-LM,[42] T5,[36] and GPT-3.[4] An important paper[47] has estimated the carbon footprint of several NLP models and argued this trend is both environmentally unfriendly and prohibitively expensive, raising barriers to participation in NLP research. We refer to such work as Red AI.

a For brevity, we refer to AI throughout this article, but our focus is on AI research that relies on deep learning methods.

This trend is driven by the strong focus of the AI community on obtaining "state-of-the-art" results,[b] as exemplified by the popularity of leaderboards,[53,54] which typically report accuracy (or other similar measures) but omit any mention of cost or efficiency (see, for example, leaderboards.allenai.org).[c] Despite the clear benefits of improving model accuracy, the focus on this single metric ignores the economic, environmental, and social cost of reaching the reported results.

We advocate increasing research activity in Green AI—AI research that is more environmentally friendly and inclusive. We emphasize that Red AI research has been yielding valuable scientific contributions to the field, but it has been overly dominant. We want to shift the balance toward the Green AI option—to ensure any inspired undergraduate with a laptop has the opportunity to write high-quality papers that could be accepted at premier research conferences. Specifically, we propose making efficiency a more common evaluation criterion for AI papers alongside accuracy and related measures.

b Meaning, in practice, that a system's accuracy on some benchmark is greater than any previously reported system's accuracy.

c Some leaderboards do focus on efficiency (https://dawn.cs.stanford.edu/benchmark/).

» key insights

- **The computational costs of state-of-the-art AI research has increased 300,000x in recent years. This trend, denoted Red AI, stems from the AI community's focus on accuracy while paying attention to efficiency.**
- **Red AI leads to a surprisingly large carbon footprint, and makes it difficult for academics, students, and researchers to engage in deep learning research.**
- **An alternative is Green AI, which treats efficiency as a primary evaluation criterion alonside accuracy. To measure efficiency, we suggest reporting the number of floating-point operations required to generate a result.**
- **Green AI research will decrease AI's environmental footprint and increase its inclusivity.**

ILLUSTRATION BY LISA SHEEHAN

AI research can be computationally expensive in a number of ways, but each provides opportunities for efficient improvements; for example, papers can plot performance as a function of training set size, enabling future work to compare performance even with small training budgets. Reporting the computational price tag of developing, training, and running models is a key Green AI practice (see Equation 1). In addition to providing transparency, price tags are baselines that other researchers could improve on.

Our empirical analysis in Figure 2 suggests the AI research community has paid relatively little attention to computational efficiency. In fact, as Figure 1 illustrates, the computational cost of high-budget research is increasing exponentially, at a pace that far exceeds Moore's Law.[33] Red AI is on the rise despite the well-known diminishing returns of increased cost (for example, Figure 3).

This article identifies key factors that contribute to Red AI and advocates the introduction of a simple, easy-to-compute efficiency metric that could help make some AI research greener, more inclusive, and perhaps more cognitively plausible. Green AI is part of a broader, long-standing interest in environmentally friendly scientific research (for example, see the *Journal Green Chemistry*). Computer science, in particular, has a long history of investigating sustainable and energy-efficient computing (for example, see the *Journal Sustainable Computing: Informatics and Systems*).

In this article, we analyze practices that move deep-learning research into the realm of Red AI. We then discuss our proposals for Green AI and consider related work, and directions for future research.

Figure 1. The amount of compute used to train deep learning models has increased 300,000x in six years. Figure taken from Amodei et al.[2]

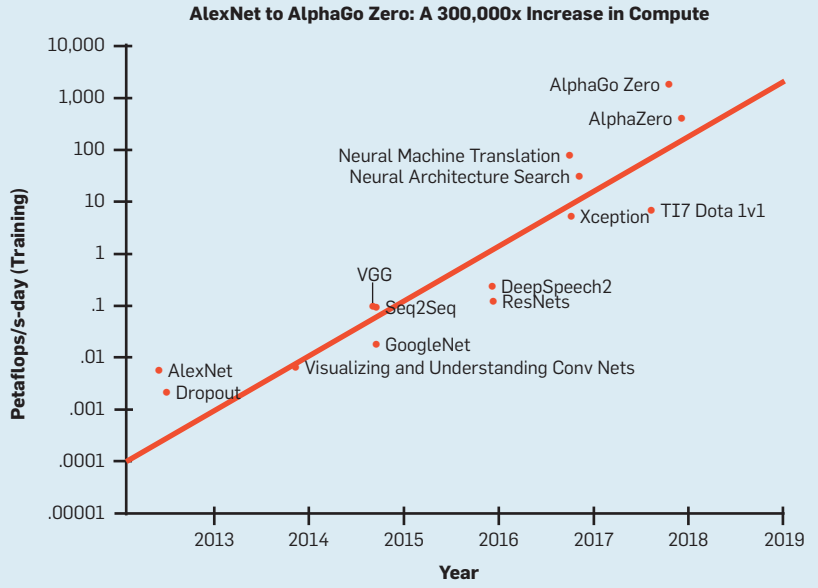
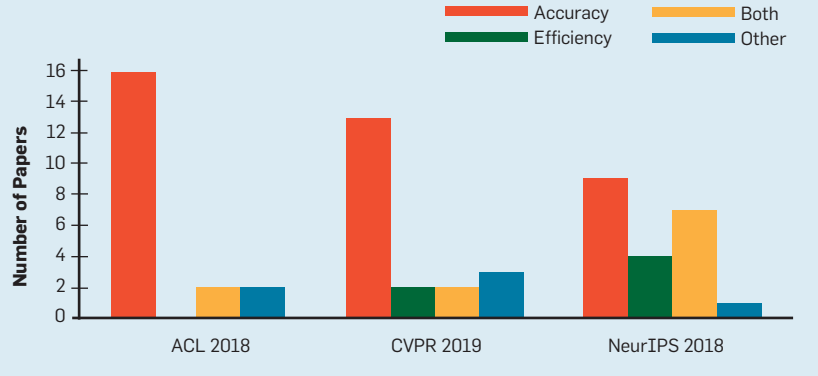


Figure 2. AI papers tend to target accuracy rather than efficiency. The figure shows the proportion of papers that target accuracy, efficiency, both or other from a random sample of 60 papers from top AI conferences.



Red AI

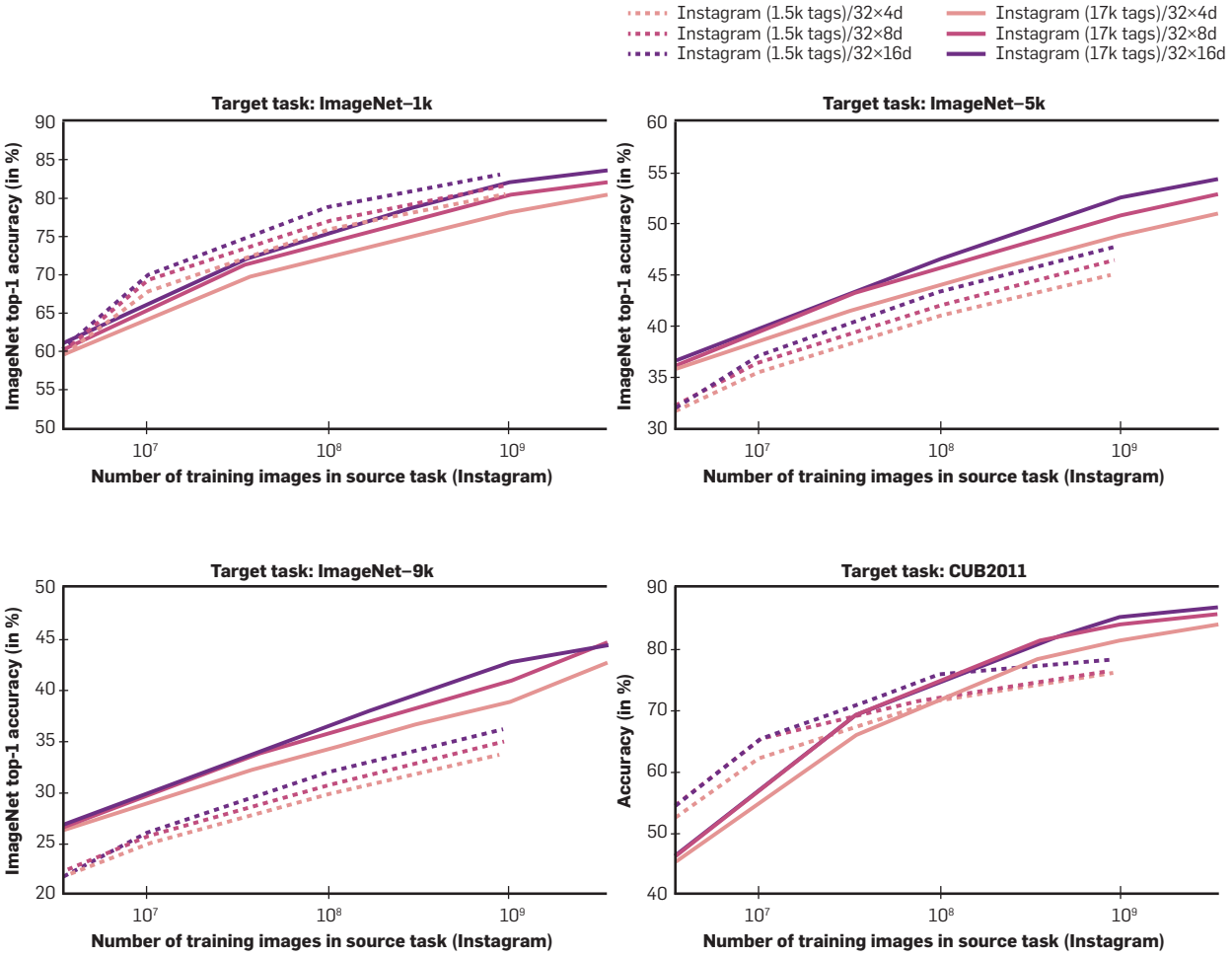
Red AI refers to AI research that seeks to improve accuracy (or related measures) through the use of massive computational power while disregarding the cost—essentially "buying" stronger results. Yet the relationship between model performance and model complexity (measured as number of parameters or inference time) has long been understood to be at best logarithmic; for a linear gain in performance, an exponentially larger model is required.[20] Similar trends exist with increasing the quantity of training data[14,48] and the number of experiments.[9,10] In each of these cases, diminishing returns come at increased computational cost.

This section analyzes the factors contributing to Red AI and shows how it is resulting in diminishing returns over time (see Figure 3). We note that Red AI work is valuable, and in fact, much of it contributes to what we know by pushing the boundaries of AI. Our exposition here is meant to highlight areas where computational expense is high, and to present each as an opportunity for developing more efficient techniques.

To demonstrate the prevalence of Red AI, we randomly sampled 60 papers from top AI conferences (ACL, NeurIPS, and CVPR).[d] For each paper we noted whether the authors claim their main contribution to be (a) an improvement to accuracy or some related measure, (b) an improvement to efficiency, (c) both, or (d) other. As shown in Figure 2, in all conferences we considered, a large majority of the papers target accuracy (90% of ACL papers, 80% of NeurIPS papers and 75% of CVPR papers). Moreover, for both empirical AI conferences (ACL

d https://acl2018.org; https://nips.cc/Conferences/2018; and http://cvpr2019.thecvf.com.

Figure 3. Diminishing returns of training on more data: object detection accuracy increases linearly as the number of training examples increases exponentially.[30]



and CVPR) only a small portion (10% and 20% respectively) argue for a new efficiency result.[e] This highlights the focus of the AI community on measures of performance such as accuracy, at the expense of measures of efficiency such as speed or model size. In this article, we argue that a larger weight should be given to the latter.

To better understand the different ways in which AI research can be red, consider an AI result reported in a scientific paper. This result typically characterizes a model trained on a training dataset and evaluated on a test dataset, and the process of developing that model often involves multiple experiments to tune its hyperparameters. We thus consider three dimensions that capture much of the computational cost of obtaining such a result: the cost of executing the model on a single (*E*)xample (either during training or at inference time); the size of the training (*D*)ataset, which controls the number of times the model is executed during training, and the number of (*H*)yperparameter experiments, which controls how many times the model is trained during model development. The total cost of producing a (*R*)esult in machine learning increases linearly with each of these quantities. This cost can be estimated as follows:

$$Cost\,(R) \propto E \cdot D \cdot H$$

Equation 1. The equation of Red AI: The cost of an AI (*R*)esult grows linearly with the cost of processing a single (*E*)xample, the size of the training (*D*)ataset and the number of (*H*)yperparameter experiments.

Equation 1 is a simplification (for example, different hyperparameter assignments can lead to different costs for processing a single example). It also ignores other factors such as the number of training epochs or data augmentation. Nonetheless, it illustrates three quantities that are each an important factor in the total cost of generating a result. Next, we consider each quantity separately.

Expensive processing of one example. Our focus is on neural models, where it

e Interestingly, many NeurIPS papers included convergence rates or regret bounds that describe performance as a function of examples or iterations, thus targeting efficiency (55%). This indicates an increased awareness of the importance of this concept, at least in theoretical analyses.

is common for each training step to require inference, so we discuss training and inference cost together as "processing" an example (though see discussion below). Some works have used increasingly large models in terms of, for example, model parameters, and as a result, in these models, performing inference can require a lot of computation, and training even more so. For instance, Google's BERT-large[8] contains roughly 350 million parameters. OpenAI's openGPT2-XL model[35] contains 1.5 billion parameters. AI2, our home organization, released Grover,[57] also containing 1.5 billion parameters. NVIDIA released Megatron-LM,[42] containing over 8 billion parameters. Google's T5-11B[36] contains 11 billion parameters. Most recently, openAI released openGPT-3,[4] containing 175 billion parameters. In the computer vision community, a similar trend is observed (Figure 1).

Such large models have high costs for processing each example, which leads to large training costs. BERT-large was trained on 64 TPU chips for four days at an estimated cost of $7,000. Grover was trained on 256 TPU chips for two weeks, at an estimated cost of $25,000. XLNet had a similar architecture to BERT-large, but used a more expensive objective function (in addition to an order of magnitude more data), and was trained on 512 TPU chips for 2.5 days, costing more than $60,000.[f] It is impossible to reproduce the best BERT-large results or XLNet results using a single GPU,[g] and models such as openGPT2 are too large to be used in production.[h] Specialized models can have even more extreme costs, such as AlphaGo, the best version of which required 1,920 CPUs and 280 GPUs to play a single game of Go,[44] with an estimated cost to reproduce this experiment of $35,000,000.[i,j]

When examining variants of a single model (for example, BERT-small and BERT-large) we see that larger models can have stronger performance, which is a valuable scientific contribution. However, this implies the financial and environmental cost of increasingly large AI models will not decrease soon, as the pace of model growth far exceeds the resulting increase in model performance.[18] As a result, more and more resources are going to be required to keep improving AI models by simply making them larger.

Finally, we note that in some cases the price of processing one example might be different at training and test time. For instance, some methods target efficient inference by learning a smaller model based on the large trained model. These models often do not lead to more efficient training, as the cost of E is only reduced at inference time. Models used in production typically have computational costs dominated by inference rather than training, but in research training is typically much more frequent, so we advocate studying methods for efficient processing of one example in both training and inference.

Processing many examples. Increased amounts of training data have also contributed to progress in state-of-the-art performance in AI. BERT-large had top performance in 2018 across many NLP tasks after training on three billion word-pieces. XLNet outperformed BERT after training on 32 billion word-pieces, including part of Common Crawl; openGPT-2-XL trained on 40 billion words; FAIR's RoBERTa[28] was trained on 160GB of text, roughly 40 billion word-pieces, requiring around 25,000 GPU hours to train. T5-11B[36] was trained on 1 trillion tokens, 300 times more than BERT-large. In computer vision, researchers from Facebook[30] pretrained an image classification model on 3.5 billion images from Instagram, three orders of magnitude larger than existing labeled image datasets such as Open Images.[k]

The use of massive data creates barriers for many researchers to reproducing the results of these models, and to training their own models on the same setup (especially as training for multiple epochs is standard). For example, the July 2019 Common Crawl contains 242TB of uncompressed data,[l] so even storing the data is expensive. Finally, as in the case of model size, relying on more data to improve performance is notoriously expensive because of the diminishing returns of adding more data.[48] For instance, Figure 3, taken from Mahajan et al.,[30] shows a logarithmic relation between the object recognition top-1 accuracy and the number of training examples.

Massive number of experiments. Some projects have poured large amounts of computation into tuning hyperparameters or searching over neural architectures, well beyond the reach of most researchers. For instance, researchers from Google[59] trained over 12,800 neural networks in their neural architecture search to improve performance on object detection and language modeling. With a fixed architecture, researchers from DeepMind[31] evaluated 1,500 hyperparameter assignments to demonstrate that an LSTM language model[17] can reach state-of-the-art perplexity results. Despite the value of this result in showing that the performance of an LSTM does not plateau after only a few hyperparameter trials, fully exploring the potential of other competitive models for a fair comparison is prohibitively expensive.

The value of massively increasing the number of experiments is not as well studied as the first two discussed previously. In fact, the number of experiments performed during model construction is often underreported. Nonetheless, evidence for a logarithmic relation exists here as well.[9,10]

Discussion. The increasing costs of AI experiments offer a natural economic motivation for developing more efficient AI methods. It might be the case that at a certain point prices will be too high, forcing even researchers with large budgets to develop more efficient methods. Our analysis in Figure 2 shows that currently most effort is still being dedicated to accuracy rather than efficiency. At the same time, AI technology is already very expensive to train or execute, which limits the ability of many researchers to study it, and of practitioners to adopt it. Combined with environmental pricetag of AI,[47] we believe more effort should be devoted toward efficient AI solutions.

f https://syncedreview.com/2019/06/27/the-staggering-cost-of-training-sota-aimodels/

g See https://github.com/google-research/bert and https://github.com/zihangdai/xlnet.

h https://towardsdatascience.com/too-big-to-deploy-how-gpt-2-is-breakingproduction-63ab29f0897c

i https://www.yuzeh.com/data/agz-cost.html

j Recent versions of AlphaGo are far more efficient.[46]

k https://opensource.google.com/projects/open-images-dataset

l http://commoncrawl.org/2019/07/

We want to reiterate that Red AI work is extremely valuable, and in fact, much of it contributes to what we know about pushing the boundaries of AI. Indeed, there is value in pushing the limits of model size, dataset size, and the hyperparameter search budget.

In addition, Red AI can provide opportunities for future work to promote efficiency; for example, evaluating a model on varying amounts of training data will provide an opportunity for future researchers to build on the work without needing a budget large enough to train on a massive dataset. Currently, despite the massive amount of resources put into recent AI models, such investment still pays off in terms of downstream performance (albeit at an increasingly lower rate). Finding the point of saturation (if such exists) is an important question for the future of AI. Moreover, Red AI costs can even sometimes be amortized, because a Red AI trained module may be reused by many research projects as a built-in component, which doesn't require retraining.

The goal of this article is twofold: first, we want to raise awareness to the cost of Red AI and encourage researchers that use such methods to take steps to allow for more equitable comparisons, such as reporting training curves. Second, we want to encourage the AI community to recognize the value of work by researchers that take a different path, optimizing efficiency rather than accuracy. Next, we turn to discuss concrete measures for making AI more green.

GREEN AI

The term Green AI refers to AI research that yields novel results while taking into account the computational cost, encouraging a reduction in resources spent. Whereas Red AI has resulted in rapidly escalating computational (and thus carbon) costs, Green AI promotes approaches that have favorable performance/efficiency trade-offs. If measures of efficiency are widely accepted as important evaluation metrics for research alongside accuracy, then researchers will have the option of focusing on the efficiency of their models with positive impact on both inclusiveness and the environment. Here, we review several measures of efficiency that could be reported and optimized, and advocate one particular measure—FPO—which we argue should be reported when AI research findings are published.

Some projects have poured large amounts of computation into tuning hyperparameters or searching over neural architectures, well beyond the reach of most researchers.

Measures of efficiency. To measure efficiency, we suggest reporting the amount of work required to generate a result. Specifically, the amount of work required to train a model, and if applicable, the aggregated amount of work required for all hyperparameter tuning experiments. As the cost of an experiment decomposes into the cost of a processing a single example, the size of the dataset, and the number of experiments (Equation 1), reducing the amount of work in each of these steps will result in AI that is more green.

We do encourage AI practitioners to use efficient hardware to reduce energy costs, but the dramatic increase in computational cost observed over recent years is primarily from modeling and algorithmic choices; our focus is on how to incorporate efficiency there. When reporting the amount of work done by a model, we want to measure a quantity that allows for a fair comparison between different models. As a result, this measure should ideally be stable across different labs, at different times, and using different hardware.

Carbon emission. Carbon emission is appealing as it is a quantity we want to directly minimize. Nonetheless it is difficult to measure the exact amount of carbon released by training or executing a model, and accordingly—generating an AI result, as this amount depends highly on the local electricity infrastructure (though see initial efforts by Henderson et al.[16] and Lacoste et al.[25]). As a result, it is not comparable between researchers in different locations or even the same location at different times.[16]

Electricity usage. Electricity usage is correlated with carbon emission while being time- and location-agnostic. Moreover, GPUs often report the amount of electricity each of their cores consume at each time point, which facilitates the estimation of the total amount of electricity consumed by generating an AI result. Nonetheless, this measure is hardware dependent, and as a result does not allow for a fair comparison between different models developed on different machines.

Elapsed real time. The total running time for generating an AI result is a natural measure for efficiency, as all other

things being equal, a faster model is doing less computational work. Nonetheless, this measure is highly influenced by factors such as the underlying hardware, other jobs running on the same machine, and the number of cores used. These factors hinder the comparison between different models, as well as the decoupling of modeling contributions from hardware improvements.

Number of parameters. Another common measure of efficiency is the number of parameters (learnable or total) used by the model. As with runtime, this measure is correlated with the amount of work. Unlike the other measures described previously, it does not depend on the underlying hardware. Moreover, this measure also highly correlates with the amount of memory consumed by the model. Nonetheless, different algorithms make different use of their parameters, for instance by making the model deeper vs. wider. As a result, different models with a similar number of parameters often perform different amounts of work.

FPO. As a concrete measure, we suggest reporting the total number of floating-point operations (FPO) required to generate a result.[m] FPO provides an estimate of the amount of work performed by a computational process. It is computed analytically by defining a cost to two base operations, ADD and MUL. Based on these operations, the FPO cost of any machine learning abstract operation (for example, a tanh operation, a matrix multiplication, a convolution operation, or the BERT model) can be computed as a recursive function of these two operations. FPO has been used in the past to quantify the energy footprint of a model,[13,32,50,51] but is not widely adopted in AI. FPO has several appealing properties. First, it directly computes the amount of work done by the running machine when executing a specific instance of a model and is thus tied to the amount of energy consumed. Second, FPO is agnostic to the hardware on which the model is run. This facilitates fair comparisons between different approaches, unlike the measures described above. Third, FPO is often correlated with the running time of the model[5] (though see discussion below). Unlike asymptotic runtime, FPO also considers the amount of work done at each time step.

The term Green AI refers to AI research that yields novel results while taking into account the computational cost, encouraging a reduction in resources spent.

Several packages exist for computing FPO in various neural network libraries,[n] though none of them contains all the building blocks required to construct all modern AI models. We encourage the builders of neural network libraries to implement such functionality directly.

Discussion. Efficient machine learning approaches have received attention in the research community but are generally not motivated by being green. For example, a significant amount of work in the computer vision community has addressed efficient inference,[13,38,58] which is necessary for real-time processing of images for applications like self-driving cars,[27,29,37] or for placing models on devices such as mobile phones.[18,40] Most of these approaches only minimize the cost of processing a single example, while ignoring the other two red practices discussed perviously.[o] Other methods to improve efficiency aim to develop more efficient architectures, starting from the adoption of graphical processing units (GPU) to AI algorithms, which was the driving force behind the deep learning revolution, up to more recent development of hardware such as tensor processing units (TPUs[22]).

The examples here indicate the path to making AI green depends on how it is used. When developing a new model, much of the research process involves training many model variants on a training set and performing inference on a small development set. In such a setting, more efficient training procedures can lead to greater savings, while in a production setting more efficient inference can be more important. We advocate for a holistic view of computational savings which doesn't sacrifice in some areas to make advances in others.

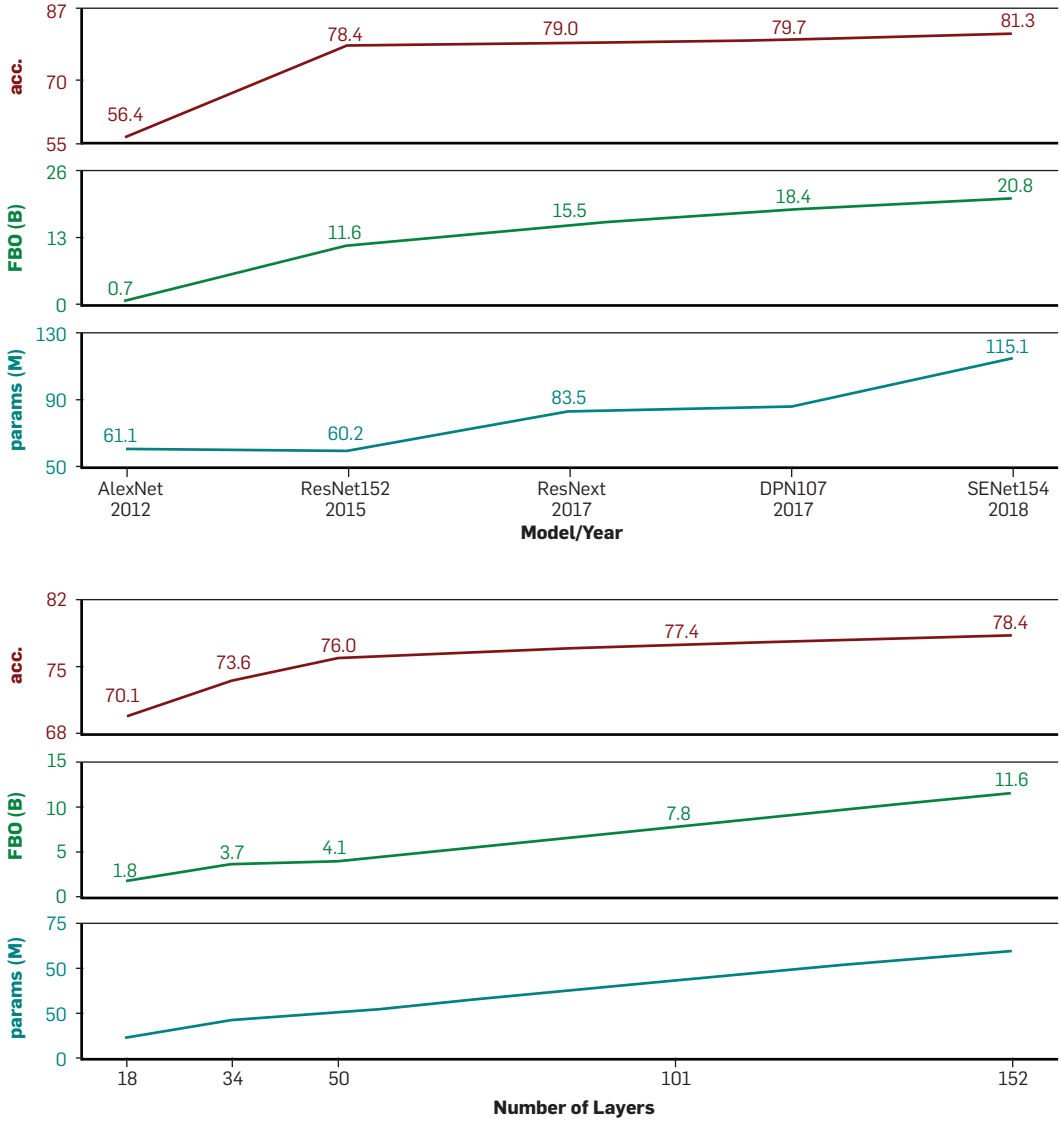
FPO has some limitations. Most importantly, the energy consumption of a

m Floating point operations are often referred to as FLOP(s), though this term is not uniquely defined.[13] To avoid confusion, we use the term FPO.

n For example, https://github.com/Swall0w/torchstat; https://github.com/Lyken17/pytorch-OpCounter

o In fact, creating smaller models often results in longer running time, so mitigating the different trends might be at odds.[52]

Figure 4. Increase in FPO leads to diminishing return for object detection top-1 accuracy. Plots (bottom to top): model parameters (in million), FPO (in billions), top-1 accuracy on ImageNet. 4(a). Leading object recognition models: AlexNet,[24] ResNet,[15] ResNext,[55] DPN107,[6] SENet154.[19] 4(b): Comparison of different sizes (measured by the number of layers) of the ResNet model.[15]



model is not only influenced by the amount of work, but also from other factors such as the communication between the different components, which is not captured by FPO. As a result, FPO doesn't always correlate with other measures such as runtime[21] and energy consumption.[16] Second, FPO targets the number of operations performed by a model, while ignoring other potential limiting factors for researchers such as the memory used by the model, which can often lead to additional energy and monetary costs.[29] Finally, the amount of work done by a model largely depends on the model implementation, as two different implementations of the same model could result in very different amounts of processing work. Due to the focus on the modeling contribution, the AI community has traditionally ignored the quality or efficiency of models' implementation.[p] We argue the time to reverse this norm has come, and that exceptionally good implementations that lead to efficient models should be credited by the AI community.

FPO cost of existing models. To demonstrate the importance of reporting the amount of work, we present FPO costs for several existing models.[q] Figure 4(a) shows the number of parameters and FPO of several leading object recognition models, as well as their performance on the ImageNet

p We consider this exclusive focus on the final prediction another symptom of Red AI.

q These numbers represent FPO per inference, that is, the work required to process a single example.

dataset.[7,r] A few trends are observable. First, as discussed earlier, models get more expensive with time, but the increase in FPO does not lead to similar performance gains. For instance, an increase of almost 35% in FPO between ResNet and ResNext (second and third points in graph) resulted in a 0.5% top-1 accuracy improvement. Similar patterns are observed when considering the effect of other increases in model work. Second, the number of model parameters does not tell the whole story: AlexNet (first point in the graph) actually has more parameters than ResNet (second point), but dramatically less FPO, and also much lower accuracy.

Figure 4(b) shows the same analysis for a single object recognition model, ResNet,[15] while comparing different versions of the model with different numbers of layers. This creates a controlled comparison between the different models, as they are identical in architecture, except for their size (and accordingly, their FPO cost). Once again, we notice the same trend: the large increase in FPO cost does not translate to a large increase in performance.

Additional ways to promote Green AI. There are many ways to encourage research that is more green. In addition to reporting the FPO cost for each term in Equation 1, we encourage researchers to report budget/performance curves where possible. For example, training curves provide opportunities for future researchers to compare at a range of different budgets and running experiments with different model sizes provides valuable insight into how model size impacts performance. In a recent paper,[9] we observed that the claim as to which model performs best depends on the computational budget available during model development. We introduced a method for computing the expected best validation performance of a model as a function of the given budget. We argue that reporting this curve will allow users to make wiser decisions about their selection of models and highlight the stability of different approaches.

We further advocate for making efficiency an official contribution in major AI conferences by advising reviewers to recognize and value contributions that do not strictly improve state of the art but have other benefits such as efficiency. Finally, we note that the trend of releasing pretrained models publicly is a green success, and we would like to encourage organizations to continue to release their models in order to save others the costs of retraining them.

Related Work

Recent work has analyzed the carbon emissions of training deep NLP models[47] and concluded that computationally expensive experiments can have a large environmental and economic impact. With modern experiments using such large budgets, many researchers (especially those in academia) lack the resources to work in many high-profile areas; increased value placed on computationally efficient approaches will allow research contributions from more diverse groups. We emphasize that the conclusions of Stubell et al.[47] are the result of long-term trends, and are not isolated within NLP, but hold true across machine learning.

While some companies offset electricity usage by purchasing carbon credits, it is not clear that buying credits is as effective as using less energy. In addition, purchasing carbon credits is voluntary; Google cloud[s] and Microsoft Azure[t] purchase carbon credits to offset their spent energy, but Amazon's AWS[u] (the largest cloud computing platform[v]) only covered 50% of its power usage with renewable energy.

The push to improve state-of-the-art performance has focused the research community's attention on reporting the single best result after running many experiments for model development and hyperparameter tuning. Failure to fully report these experiments prevents future researchers from understanding how much effort is required to reproduce a result or extend it.[9]

Our focus is on improving efficiency in the machine learning community, but machine learning can also be used as a tool for work in areas like climate change. For example, machine learning has been used for reducing emissions of cement plants[1] and tracking animal conservation outcomes,[12] and is predicted to be useful for forest fire management.[39] Undoubtedly these are important applications of machine learning; we recognize they are orthogonal to the content of this article.

Conclusion

The vision of Green AI raises many exciting research directions that help to overcome the challenges of Red AI. Progress will find more efficient ways to allocate a given budget to improve performance, or to reduce the computational expense with a minimal reduction in performance. Also, it would seem that Green AI could be moving us in a more cognitively plausible direction as the brain is highly efficient.

It is important to reiterate that we see Green AI as a valuable option, not an exclusive mandate—of course, both Green AI and Red AI have contributions to make. Our goals are to augment Red AI with green ideas, like using more efficient training methods, and reporting training curves; and to increase the prevalence of Green AI by highlighting its benefits, advocating a standard measure of efficiency. Here, we point to a few important green research directions, and highlight a few open questions.

Research on building space- or time-efficient models is often motivated by fitting a model on a small device (such as a phone) or fast enough to process examples in real time, such as image captioning for the blind (as discussed previously). Here, we argue for a far broader approach that promotes efficiency for all parts of the AI development cycle.

Data efficiency has received significant attention over the years.[23,41,49] Modern research in vision and NLP often involves first pretraining a model on large "raw" (unannotated) data then finetuning it to a task of interest through supervised learning. A strong result in this area often involves achieving similar performance to a baseline with fewer training examples or fewer gradient steps. Most recent work has addressed fine-tuning data,[34] but pretraining efficiency is also important. In either case, one simple technique to improve in this area is to

r Numbers taken from https://github.com/sovrasov/flops-counter.pytorch.

s https://cloud.google.com/sustainability/

t https://www.microsoft.com/en-us/environment/carbon

u https://aws.amazon.com/about-aws/sustainability/

v https://tinyurl.com/y2kob969

simply report performance with different amounts of training data. For example, reporting performance of contextual embedding models trained on 10 million, 100 million, 1 billion, and 10 billion tokens would facilitate faster development of new models, as they can first be compared at the smallest data sizes.

Research here is of value not just to make training less expensive, but because in areas such as low resource languages or historical domains it is extremely difficult to generate more data, so to progress we must make more efficient use of what is available.

Finally, the total number of experiments run to get a final result is often underreported and underdiscussed.[9] The few instances researchers have of full reporting of the hyperparameter search, architecture evaluations, and ablations that went into a reported experimental result has surprised the community.[47] While many hyperparameter optimization algorithms exist, which can reduce the computational expense required to reach a given level of performance,[3,11] simple improvements here can have a large impact. For example, stopping training early for models that are clearly underperforming can lead to great savings.[26]

Acknowledgment. This research was conducted at the Allen Institute for AI. C

References

1. Acharyya, P., Rosario, S.D., Flor, F., Joshi, R., Li, D., Linares, R, and Zhang, H. Autopilot of cement plants for reduction of fuel consumption and emissions. In *Proceedings of ICML Workshop on Climate Change*, 2019.
2. Amodei, D. and Hernandez, D. AI and compute, 2018. Blog post.
3. Bergstra, J.S., Bardenet, R., Bengio, Y. and Kégl, B. Algorithms for hyper-parameter optimization. In *Proceedings of NeurIPS*, 2011.
4. Brown, T.B. et al. Language models are few-shot learners, 2020; arXiv:2005.14165.
5. Canziani, A., Paszke, A. and Culurciello, E. An analysis of deep neural network models for practical applications. In *Proceedings of ISCAS*, 2017.
6. Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S. and Feng, J. Dual path networks. In *Proceedings of NeurIPS*, 2017.
7. Deng, J., Dong, W., Socher, R., Li, L-J, Li, K. and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *Proceedings of CVPR*, 2009.
8. Devlin, J., Chang, M.W., Lee, K., and Toutanova, K. BERT: Pretraining of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, 2019.
9. Dodge, J., Gururangan, S., Card, D., Schwartz, R. and Smith, N.A. Show your work: Improved reporting of experimental results. In *Proceedings of EMNLP*, 2019.
10. Dodge, J., Ilharco, G., Schwartz, R., Farhadi, A., Hajishirzi, H. and Smith, N.A. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping, 2020; arXiv:2002.06305.
11. Dodge, J., Jamieson, K. and Smith, N.A. Open loop hyperparameter optimization and determinantal point processes. In *Proceedings of AutoML*, 2017.
12. Duhart, C., Dublon, G., Mayton, B., Davenport, G. and Paradiso, J.A. Deep learning for wildlife conservation and restoration efforts. In *Proceedings of ICML Workshop on Climate Change*, 2019.
13. Gordon, A., Eban, E., Nachum, O., Chen, B., Wu, H., Yang, T-J, and Choi, E. MorphNet: Fast & simple resource-constrained structure learning of deep networks. In *Proceedings of CVPR*, 2018.
14. Halevy, A., Norvig, P. and Pereira, F. The unreasonable effectiveness of data. *IEEE Intelligent Systems 24* (2009), 8–12.
15. He, K., Zhang, X., Ren, S. and Sun, J. Deep residual learning for image recognition. In *Proceedings of CVPR*, 2016.
16. Henderson, P., Hu, J., Romoff, J., Brunskill, E., Jurafsky, D. and Pineau, J. Towards the systematic reporting of the energy and carbon footprints of machine learning, 2020; arXiv:2002.05651.
17. Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Computation 9*, 8 (1997), 1735–1780.
18. Howard, A.G. et al. MobileNets: Efficient convolutional neural networks for mobile vision applications, 2017; arXiv:1704.04861.
19. Hu, J., Shen, L. and Sun, G. Squeeze-and-excitation networks. In *Proceedings of CVPR*, 2018.
20. Huang, J. et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *Proceedings of CVPR*, 2017.
21. Jeon, Y. and Kim, J. Constructing fast network through deconstruction of convolution. In *Proceedings of NeurIPS*, 2018.
22. Jouppi, N.P. et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of ISCA 1*, 1 (2017), Publ. date: June 2020.
23. Kamthe, S. and Deisenroth, M.P. Data-efficient reinforcement learning with probabilistic model predictive control. In *Proceedings of AISTATS*, 2018.
24. Krizhevsky, A., Sutskever, I. and Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Proceedings of NeurIPS*, 2012.
25. Lacoste, A., Luccioni, A., Schmidt, V. and Dandres, T. Quantifying the carbon emissions of machine learning. In *Proceedings of the Climate Change AI Workshop*, 2019.
26. Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A. and Talwalkar, A. Hyperband: Bandit-based configuration evaluation for hyperparameter optimization. In *Proceedings of ICLR*, 2017.
27. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S. Fu, C-Y and Berg, A.C. SSD: Single shot multibox detector. In *Proceedings of ECCV*, 2016.
28. Liu, Y. et al. RoBERTa: A robustly optimized BERT pretraining approach, 2019; arXiv:1907.11692.
29. Ma, N., Zhang, X., Zheng, H.T and Sun, J. ShuffleNet V2: Practical guidelines for efficient cnn architecture design. In *Proceedings of ECCV*, 2018.
30. Mahajan, D. et al. Exploring the limits of weakly supervised pretraining, 2018; arXiv:1805.00932.
31. Melis, G., Dyer, C. and Blunsom, P. On the state of the art of evaluation in neural language models. In *Proceedings of EMNLP*, 2018.
32. Molchanov, P., Tyree, S., Karras, T., Aila, T. and Kautz, J. Pruning convolutional neural networks for resource efficient inference. In *Proceedings of ICLR*, 2017.
33. Moore, G.E. Cramming more components onto integrated circuits, 1965.
34. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L. Deep contextualized word representations. In *Proceedings of NAACL*, 2018.
35. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I. Language m odels are unsupervised multitask learners.. OpenAI Blog, 2019.
36. Raffel, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019; arXiv:1910.10683.
37. Rastegari, M., Ordonez, V., Redmon, J. and Farhadi, A. Xnornet: Imagenet classification using binary convolutional neural networks. In *Proceedings of ECCV*, 2016.
38. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of CVPR*, 2016.
39. Rolnick, D. et al. Tackling climate change with machine learning, 2019; arXiv:1905.12616.
40. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A. and Chen, L.C. MobileNetV2: Inverted residuals and linear bottlenecks. In *Proceedings of CVPR*, 2018.
41. Schwartz, R., Thomson, S. and Smith, N.A. SoPa: Bridging CNNs, RNNs, and weighted finite-state machines. In *Proceedings of ACL*, 2018.
42. Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., Catanzaro, B. Megatron-LM: Training multi-billion parameter language models using GPU model parallelism, 2019; arXiv:1909.08053.
43. Shoham, Y. et al. The AI index 2018 annual report. AI Index Steering Committee, Human-Centered AI Initiative, Stanford University; http:// cdn.aiindex.org/2018/AI%20Index%202018%20Annual%20Report.pdf.
44. Silver, D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature 529*, 7587 (2016) 484.
45. Silver, D. et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm, 2017; arXiv:1712.01815.
46. Silver, D. et al. Mastering the game of Go without human knowledge. *Nature 550*, 7676 (2017), 354.
47. Strubell, E., Ganesh, A. and McCallum, A. Energy and policy considerations for deep learning in NLP. In *Proceedings of ACL*, 2019.
48. Sun, C., Shrivastava, A., Singh, S. and Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In Proceedings of ICCV, 2017.
49. Tsang, I., Kwok, J.T. and Cheung, P.M. Core vector machines: Fast SVM training on very large data sets. *JMLR 6* (Apr. 2005), 363–392.
50. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I. Attention is all you need. In *Proceedings of NeurIPS*, 2017.
51. Veniat, T. and Denoyer, L. Learning time/memory-efficient deep architectures with budgeted super networks. In *Proceedings of CVPR*, 2018.
52. Walsman, A., Bisk, Y., Gabriel, S., Misra, D., Artzi, Y., Choi, Y. and Fox, D. Early fusion for goal directed robotic vision. In *Proceedings of IROS*, 2019.
53. Wang, A. Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O. and Bowman, S.R. SuperGLUE: A stickier benchmark for general-purpose language understanding systems, 2019; arXiv:1905.00537.
54. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O. and Bowman, S.R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of ICLR*, 2019.
55. Xie, S., Girshick, R., Dollar, P., Tu, Z. and He, K. Aggregated residual transformations for deep neural networks. In *Proceedings of CVPR*, 2017.
56. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. and Le, Q.V. XLNet: Generalized autoregressive pretraining for language understanding, 2019; arXiv:1906.08237.
57. Zellers, R., Holtzman, A., Rashkin, H., Bisk, Y., Farhadi, A., Roesner, F. and Choi, Y. Defending against neural fake news, 2019; arXiv:1905.12616.
58. Zhang, X., Zhou, X., Lin, M. and Sun, J. ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of CVPR*, 2018.
59. Zoph, B. and Le, Q.V. Neural architecture search with reinforcement learning. In *Proceedings of ICLR*, 2017.

Roy Schwartz (roys@allenai.org) is Senior Lecturer at the Hebrew University of Jerusalem, Israel.

Jesse Dodge (dodgejesse@gmail.com), Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA.

Noah A. Smith (noah@allenai.org) is a professor of computer science and engineering at the University of Washington and senior research manager for the AllenNLP team at Allen Institute for AI and, Seattle, WA, USA.

Oren Etzioni (orene@allenai.org) is Chief Executive Officer of the Allen Institute for AI, and a professor of computer science at the University of Washington, Seattle, WA, USA.

Watch the authors discuss this work in the exclusive *Communications* video. https://cacm.acm.org/videos/green-ai