Zero Is All You Need (To Get Many Useful Properties)

Person 1 Person 2

A Department
A University
A Place, AS 12345
{person1,person2}@cs.auniv.edu

Abstract

Data analysts typically report empirical accuracy scores to support the efficacy of new methodology. But it has long been understood that good accuracy performance on a collection of available data sets is not enough to guarantee useful performance in real-world deployment of a method. For instance, learned models may fail to generalize well to new data sets due to issues with stability and robustness. Some methods may be too expensive in the face of limited resources (e.g., compute time, memory, or user time) for many analysts to run at all. In the present paper, we take the opposite perspective and ask: what if we considered a wide range of desiderata for data science methods excluding experimental accuracy? In this case, we find that a classic algorithm from the literature, namely an algorithm that always returns 0, achieves peak performance across a very wide range of desiderata.

1. Introduction

Researchers in artificial intelligence (AI), machine learning (ML), and data science considered broadly are concerned with developing accurate algorithms for some task — such as predicting whether a skin lesion is cancerous from an image [1, 17, 18], translating written or spoken text from one language to another [33, 35], or estimating the association between air pollution and health outcomes [20, 21]. Typically researchers will test their new methods extensively on a variety of data sets; for each data set, researchers will report a score or metric describing how accurate the results of their algorithm are. However, it is widely understood that good accuracy performance on a collection of available data sets is not sufficient to guarantee useful performance when a method is deployed in the real-world in the future [8]. For instance, learned models may fail to generalize well to new data sets due to issues with stability and robustness. Some methods may be prohibitively expensive in limited resources (e.g., compute time, memory, or user time) for many analysts to run [14].

DOI: http://dx.doi.org/10.1145/(to come)

In the present paper, we take the opposite perspective and ask: what if we considered a wide range of desiderata for AI/ML methods *except* for experimentally confirmed accuracy? We find that indeed, even when we focus on many such desiderata simultaneously, a single algorithm *can* achieve peak performance relative to essentially every other AI/ML method across essentially every desideratum we consider.

We call that method the Zero-Encoding Representation Operator (or ZERO for short). ZERO returns the number 0, no matter its input. While this algorithm has been proposed before and studied extensively in the literature, we are not aware of a systematic review bringing together a host of its desirable properties in one place. We here provide that review.

After describing additional related work (section 1.1), we detail the ZERO algorithm (and various illustrative special cases) in section 2. Then we review its desirable resource efficiency (section 3.1), reproducibility and replicability (section 3.2), robustness (section 3.3), portability and extensibility (section 3.4), and good theoretical performance (section 3.5). Though we are motivated to explore its properties beyond empirical performance, we note that authors have established its good empirical performance as well in section 3.6.

1.1 Related work

ZERO has been proposed by many authors and is so ubiquitous as to appear frequently in textbooks, though often by different names. Similar methods include constant predictors or estimators — though in these cases the constant is sometimes not equal to 0. While often the training-data mean is treated as a constant, we emphasize that such a choice has dependence on the training data while ZERO does not; this distinction will be relevant to our analysis below. Constant estimators (using the training-data mean or a true constant) have been extensively empirically evaluated across many papers. All of these methods, including ZERO, are so common as to be built into scikit-learn: `https://scikit-learn.org/stable/modules/model_evaluation.html#dummy-estimators`.

ZERO can be seen as a special case of essentially every popular AI/ML method. For instance, ZERO can be interpreted as a zero-layer neural network — or alternatively any neural network where the weights and biases in the final layer are all zero. ZERO also results from constructing a transformer whose feed-forward net has weights and biases set to 0. ZERO can be seen as a decision tree with a single (root) node, no splits, and a zero predictor at the root (or a summary over many such decision trees). When considered as a classifier, ZERO can be seen as offering a solution to zero-shot learning [29, 36] since it need not know all the classes in advance.

ZERO has been proposed for large language model tasks by Zheng et al. [41].

2. Setup and method

First, we detail the ZERO algorithm. We observe that ZERO offers a unified framework for a broad range of tasks of modern interest.

Algorithm. The ZERO algorithm takes any input and returns 0. See fig. 1 for a full implementation in Python. We illustrate with a number of special cases next; this list of tasks is not exhaustive for the range of applications where we can use the ZERO algorithm.

ZERO algorithm

```
class ZERO_algorithm():
    def __init__(self):
        # nothing happens here
        pass

    def run(self, inputs):
        # the output is always
            0
        return 0
```

Figure 1. Full Python code for the ZERO algorithm.

Regression. In supervised learning, we learn a transformation $f : \mathcal{X} \to \mathcal{Y}$ between features $X \in \mathcal{X}$ and outputs $Y \in \mathcal{Y}$ from some collection of observed data. In regression, $\mathcal{Y} = \mathbb{R}$. Formally, in regression, the ZERO algorithm uses the model $f(X) \equiv 0$.

Classification. In this supervised learning task, $\mathcal{Y}$ is a discrete, countable set. If $\mathcal{Y}$ is finite, we can identify its elements with $\{0, 1, \ldots, |\mathcal{Y}|\}$. If $\mathcal{Y}$ is infinite, we can identify its elements with the natural numbers $\{0, 1, \ldots\}$. In either case, our model can again be expressed as $f(X) \equiv 0$.

Language generation. In language generation, an algorithm returns a string after receiving a prompt string. Analogous to the classification case above, we can think of the set of all finite strings generated with a finite alphabet as a countable set. Given some ordering on this set, we can identify each string with a natural number. Then our model can again be expressed as $f(X) \equiv 0$, returning the 0th string (counting from 0). This model has previously been proposed and studied empirically by Zheng et al. [41].

Estimation. In estimation, we use a function of some collection of observed data as our guess for an unknown quantity. In this case, our ZERO algorithm uses the function identically equal to 0, across all possible observed data sets.

Uncertainty quantification. Often in the tasks above, we want to return not just our best guess but a range of likely values in the form of a confidence region. In this case, we use the ZERO algorithm to arrive at the width of our confidence region, namely 0. Since our output in the cases above is 0, our confidence region is therefore $\{0\}$.

3. Properties

We next describe many desirable properties of the ZERO algorithm.

3.1 Resource efficiency

ZERO is a resource-frugal algorithm; it beats or ties every other algorithm in (respectively) running time, memory, power, and user-time demands.

Running time. ZERO has $O(1)$ time complexity.

Memory. ZERO has $O(1)$ space complexity.

Low hardware demand. Depending on the implementation, ZERO can run on virtually any reasonable hardware, including microcontrollers. ZERO is therefore suitable for essentially any embedded system.

Power cost. The power cost of modern AI algorithms is of increasing concern [3, 9, 22, 38]. Also, embedded systems face strict power constraints. When ZERO is implemented well, no algorithm uses less power than ZERO.

User time: coding. ZERO is conceptually straightforward and easy to implement. It requires fewer than 10 lines of code. For the user who would prefer not to implement ZERO themselves, it is already available in many common software packages, including scikit-learn: https://scikit-learn.org/stable/modules/model_evaluation.html#dummy-estimators.

User time: Automation. Many modern AI and ML methods can require the user to choose free (tuning) parameters or user-configurable options [10, 11, 40]. While various works aim to increase the automation of otherwise-tedious tuning and hyperparameter search [12, 19, 32, 39], full automation remains elusive. ZERO, however, is fully black box; it requires no input from the user beyond choosing to run the algorithm.

3.2 Reproducibility and Replicability

While it is typically difficult to compare the reproducibility or replicability of algorithms decisively, we feel confident stating that no algorithm is more reproducible than ZERO.

Reproducibility. In their Conclusion 3-1, the National Academies of Sciences, Engineering, and Medicine [26] define *reproducibility* as "obtaining consistent results using the same input data; computational steps, methods, and code; and conditions of analysis." ZERO always gives the same results.

Facilitates debugging. Since correct implementation impacts reproducibility, we note that ZERO is straightforward to debug; if the code does anything other than output 0, rewriting the code to only output 0 (and take no other action) eliminates all bugs.

Facilitates transparency. The complexity of code and high compute demands of modern AI and ML have sometimes hindered transparency and reproducibility [13]. By contrast, it is easy to share the full algorithm and code for ZERO. Anyone with any computing resources has the resources to run and check ZERO.

Privacy-preserving. Even when run on highly sensitive data, sharing the ZERO algorithm is guaranteed to preserve the privacy of that data. This property not only offers peace of mind in a variety of sensitive applications but also facilitates wide sharing and checking of ZERO by researchers, without need for additional privacy safeguards.

Replicability. In their Conclusion 3-1, the National Academies of Sciences, Engineering, and Medicine [26] further define *replicability* as "obtaining consistent results across studies aimed at answering the same scientific question, each of which has obtained its own data." We performed an informal experiment where each of the three authors of the present paper gathered their own data, wrote their own implementation of ZERO, and ran their code on their own data. Every author arrived at exactly the same result (namely, 0). We conclude that data analyses using ZERO are highly replicable.

The reproducibility and replicability of ZERO contrast with many modern methods in AI and ML [4, 8], due in part to the aspects of stability we describe next and also in the robustness section section 3.3 below.

Randomness in training. Training and evaluation of state-of-the-art machine learning models often involve randomness [31]. For example, models trained with stochastic gradient descent can vary substantially across runs [4]. By contrast, our model is entirely deterministic; for any given fixed dataset, any two runs of the model will produce identical results.

Invariance to preprocessing. Preprocessing is crucial for the performance of many machine learning models. Data are often normalized to specific ranges, centered, or adjusted for contrast in the case of images [2, 6, 28]. However, two researchers might reason-

ably choose somewhat different preprocessing methods, and we might hope to reach the same conclusions across reasonable preprocessing choices. ZERO is entirely invariant to preprocessing. This stability holds for any preprocessing pipeline, whether reasonable or unreasonable. Given this invariance, researchers can save valuable time by entirely avoiding potentially tedious data cleaning and preprocessing procedures in advance of using ZERO.

3.3 Robustness

We might be concerned about the generalizability of our conclusions if our AI or ML method gives substantively different answers when the inputs are changed in ways we think should not affect the conclusions of our data analysis. Robustness in general is not a monolith; there are many forms of robustness [5, 7, 23, 25], and we cover only a small subset below. However, ZERO is special among algorithms in that it is robust to just about any perturbation of interest to AI and ML.

Robustness to missing data and NaNs. Missing data and NaNs are ubiquitous in real-life applications, but standard ML models often struggle in these cases [27]. ZERO, however, seamlessly handles missing data and NaNs in any part of the data. And ZERO requires no modification from the user to do so.

Robustness to adversaries. Szegedy et al. [34] demonstrated that nearly imperceptible changes to an image can lead to a substantial change in the output of an image classifier. Based on this observation, Szegedy et al. [34] and Rauber et al. [30] formalize robustness to adversarial attacks as the minimal perturbation required to change the model's output significantly.

Definition 1 (Adversarial robustness, Rauber et al. [30])**.** *For distance $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ and distance $d_x : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ and a adversary threshold $t > 0$, we can define the adversarial robustness of model g, $R_{f,t}$ to be*

$$R_{g,t} := \mathbb{E}_X \left[\min_{\delta : \ell(g(X+\delta), g(X)) \geq t} d_x(X, X+\delta) \right] \tag{1}$$

with convention that $\min_\emptyset d = \infty$.

The following proposition establishes that the ZERO model has infinite adversarial robustness, the highest one can achieve.

Proposition 1. *The ZERO model has adversarial robustness $R_{f,t} = \infty$ for all $t > 0$.*

Proof. Since for all δ and X we have $f(X+\delta) = f(X)$ and in turn $\{\delta : \ell(g(X+\delta), g(X)) \geq t\} = \emptyset$. □

It follows that the ZERO algorithm is completely impervious to adversarial attack.

Robustness to gross errors. In estimation problems, some models can also be viewed as operators on a distribution, such as the empirical distribution of a dataset. We can define the ZERO model as an operator F acting on the empirical distribution of data X, denoted as P_X. By definition, we have $F(P_X) \equiv 0$. Another measure of robustness is through the influence function, which measures the change in the operator when a small perturbation is introduced to the distribution on which it operates.

Definition 2 (Influence function and gross-error sensitivity, Huber and Ronchetti [16])**.** *Denote an $\epsilon \in [0, 1]$ contamination with a gross outlier at x of P as $P_{\epsilon,x}$, where*

$$P_{\epsilon,x} := (1-\epsilon)P + \epsilon\delta_x \tag{2}$$

and δ_x is a point mass at x. The influence function of an operator T at distribution P is defined as

$$IF_{x,T,P} = \lim_{\epsilon \to 0} \frac{T(P_{\epsilon,x}) - T(P)}{\epsilon}, \tag{3}$$

and the gross-error sensitivity can be defined as

$$\gamma^*(T, P) = \sup_x |IF_{x,T,P}|. \tag{4}$$

The ZERO model is fully robust for this notion of robustness.

Proposition 2. *The ZERO model F has 0 gross-error sensitivity.*

Proof. For all P, we have $F(P) = 0$. Thus, for all x, we have $\text{IF}_{x,F,P} = 0$. So, for all P, $\gamma^*(F, P) = 0$. □

3.4 Portability and extensibility

Software dependencies. Modern AI/ML methods often have extensive software dependencies; these have sometimes served as an impediment to wider use of new methodologies [24]. For instance, software package versions can change, with deleterious effects for an analysis [15]. ZERO, however, has no package dependencies and is trivially portable across programming languages.

Streaming data. Often new data becomes available for a problem after an original AI/ML analysis has concluded. ZERO is able to handle streaming data without modification; it is able to take in new data on the fly. It requires no knowledge in advance of the rate or ultimate size of the data stream.

Different data formats. A wide variety of data formats are collected for the purposes of data analysis — from tabular data to different image formats to differen text formats. Moveover, sometimes new data is collected in a somewhat different format than old data; e.g., somewhat different features may be collected for data points at different points in time. Different units or dimensions may be used. Different precisions may be used when recording data. Sometimes researchers need to merge data sets collected under different circumstances. ZERO is inherently able all data formats and indeed inconsistent data formats across data points.

Multimodality. A longstanding challenge in AI/ML is effectively synthesizing information across multiple image modes — such as text, images, and tabular data — in a single analysis. ZERO is able to handle any collection of diverse data types simultaneously.

3.5 Theoretical Performance

Theory is often useful to understand how and why empirical performance of AI/ML methods might be expected to generalize beyond the particular data sets considered in a data analysis. While we consider some theoretical qualities (such as robustness) above, we consider additional properties of ZERO next.

Admissibility. Intuitively admissibility of an estimator means that there does not exist another estimator that is better (in some loss) across all possible values of the parameter of interest. It is well known that, under appropriate assumptions, ZERO as an estimator is admissible [e.g., a trivial modification of 37, Example 12.18]. We review one such set of assumptions in proposition 3.

Proposition 3. *Consider data X. Suppose the distribution of X is indexed by parameter $\theta \in \Theta$ with $0 \in \Theta$ and all allowed distributions are absolutely continuous with respect to the one with $\theta = 0$. Consider a loss $L(\theta, \hat{\theta}(X)) \geq 0$ that is 0 only when $\hat{\theta}(X) = \theta$, with $\theta \in \Theta$. Then the ZERO model $f(X)$ is admissible; that is, there does not exist an estimator $\hat{\theta}$ such that, for all $\theta \in \Theta$, $\mathbb{E}_{X|\theta}(L(\theta, \hat{\theta}(X))) \leq \mathbb{E}_{X|\theta}(L(\theta, f(X)))$ and such that the inequality is strict for some θ.*

Proof. Suppose there exist such an estimator $\hat{\theta}$. When $\theta = 0$, we have $\mathbb{E}_{X|0}(L(0, f(X))) = 0$, and thus $\mathbb{E}_{X|0}(L(0, \hat{\theta}(X))) \leq 0$. Since $L \geq 0$, we must have $L(0, \hat{\theta}(X)) = 0$ almost surely. So

$\hat{\theta}(X) = 0$ almost surely under $\theta = 0$. By assumption, we have allowed all data distributions that are absolutely continuous with respect to the one indexed by $\theta = 0$. So we must also have $\hat{\theta}(X) = 0$ almost surely under any $\theta \in \Theta$. Because $\hat{\theta}(X) = 0$ almost surely for all θ the risk $\mathbb{E}_{X|\theta}(L(\theta, \hat{\theta}(X))) = \mathbb{E}_{X|\theta}(L(\theta, 0)) = \mathbb{E}_{X|\theta}(L(\theta, f(X)))$ for all θ. This observation contradicts the assumption that for some $\theta \neq 0$ with $\mathbb{E}_{X|\theta}(L(\theta, \hat{\theta}(X))) < \mathbb{E}_{X|\theta}(L(\theta, f(X)))$. □

Width of uncertainty intervals. Uncertainty intervals (such as confidence intervals or credible intervals) can summarize uncertainty around predictions or estimates. Tighter uncertainty intervals are generally preferable over wider uncertainty intervals; smaller intervals reflect that we are more certain of our result. Per section 2, the uncertainty intervals returned by ZERO have width 0. So, provably, no other uncertainty intervals can have a smaller width.

3.6 Empirical Performance

Though we were originally motivated to consider properties of ZERO separate from empirical performance, we also note that ZERO has sometimes been seen to exhibit state-of-the-art empirical performance.

Natural language processing. In language generation, ZERO can be seen as equivalent to the NullModel tested by Zheng et al. [41]; see section 2. Zheng et al. [41] demonstrated that their NullModel achieves a high winning rate in automatic large language model evaluation pipelines.

4. Discussion

We have seen in the course of this review that ZERO has an extremely wide range of desirable properties for AI/ML methods. But it is also self-evidently a very bad output in essentially every AI/ML task — from prediction to estimation to language or image generation. This juxtaposition serves as a reminder that these properties, even all in concert, do not suffice to define a good AI/ML output. Moreover, some of these properties may come with a cost. Careful considerations of potential trade-offs between robustness, efficiency, and other considerations during model development must be taken into account.

References

[1] A. Adegun and S. Viriri. Deep learning techniques for skin lesion analysis and melanoma cancer detection: a survey of state-of-the-art. *Artificial Intelligence Review*, 54(2):811–841, 2021.

[2] B. Bala and S. Behal. A brief survey of data preprocessing in machine learning and deep learning techniques. In *2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 1755–1762. IEEE, 2024.

[3] N. Bashir, P. Donti, J. Cuff, S. Sroka, M. Ilic, V. Sze, C. Delimitrou, and E. Olivetti. The climate and sustainability implications of generative ai. 2024.

[4] A. L. Beam, A. K. Manrai, and M. Ghassemi. Challenges to the reproducibility of machine learning models in health care. *Jama*, 323 (4):305–306, 2020.

[5] T. Broderick, A. Gelman, R. Meager, A. L. Smith, and T. Zheng. Toward a taxonomy of trust for probabilistic machine learning. *Science advances*, 9(7):eabn3999, 2023.

[6] J. Chaki and N. Dey. *A beginner's guide to image preprocessing techniques*. CRC Press, 2018.

[7] B. Chander, C. John, L. Warrier, and K. Gopalakrishnan. Toward trustworthy artificial intelligence (tai) in the context of explainability and robustness. *ACM Computing Surveys*, 57(6):1–49, 2025.

[8] A. D'Amour, K. A. Heller, D. Moldovan, B. Adlam, B. Alipanahi, A. Beutel, C. Chen, J. Deaton, J. Eisenstein, M. D. Hoffman, F. Hormozdiari, N. Houlsby, S. Hou, G. Jerfel, A. Karthikesalingam, M. Lucic, Y. Ma, C. Y. McLean, D. Mincu, A. Mitani, A. Montanari, Z. Nado, V. Natarajan, C. Nielson, T. F. Osborne, R. Raman, K. Ramasamy, R. Sayres, J. Schrouff, M. Seneviratne, S. Sequeira, H. Suresh, V. Veitch, M. Vladymyrov, X. Wang, K. Webster, S. Yadlowsky, T. Yun, X. Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, 23(226):1–61, 2022. URL http://jmlr.org/papers/v23/20-1335.html.

[9] A. De Vries. The growing energy footprint of artificial intelligence. *Joule*, 7(10):2191–2194, 2023.

[10] J. Elith, J. R. Leathwick, and T. Hastie. A working guide to boosted regression trees. *Journal of animal ecology*, 77(4):802–813, 2008.

[11] M. Feurer and F. Hutter. *Hyperparameter optimization*. Springer International Publishing, 2019.

[12] R. Giordano, M. Ingram, and T. Broderick. Black box variational inference with a deterministic objective: Faster, more accurate, and even more black box. *Journal of Machine Learning Research*, 25(18): 1–39, 2024.

[13] B. Haibe-Kains, G. A. Adam, A. Hosny, F. Khodakarami, M. A. Q. C. M. S. B. of Directors Shraddha Thakkar 35 Kusko Rebecca 36 Sansone Susanna-Assunta 37 Tong Weida 35 Wolfinger Russ D. 38 Mason Christopher E. 39 Jones Wendell 40 Dopazo Joaquin 41 Furlanello Cesare 42, L. Waldron, B. Wang, C. McIntosh, A. Goldenberg, A. Kundaje, et al. Transparency and reproducibility in artificial intelligence. *Nature*, 586(7829):E14–E16, 2020.

[14] W. D. Heaven. Ai is wrestling with a replication crisis. *MIT Technology Review*, (12 Nov), 2020.

[15] B. J. Heil, M. M. Hoffman, F. Markowetz, S.-I. Lee, C. S. Greene, and S. C. Hicks. Reproducibility standards for machine learning in the life sciences. *Nature Methods*, 18:1132–1135, 2021.

[16] P. J. Huber and E. M. Ronchetti. *Robust statistics*. John Wiley & Sons, 2011.

[17] O. Jones, R. Matin, M. Van der Schaar, K. P. Bhayankaram, C. Ranmuthu, M. Islam, D. Behiyat, R. Boscott, N. Calanzani, J. Emery, et al. Artificial intelligence and machine learning algorithms for early detection of skin cancer in community and primary care settings: a systematic review. *The Lancet Digital Health*, 4(6):e466–e476, 2022.

[18] M. A. Kassem, K. M. Hosny, R. Damaševičius, and M. M. Eltoukhy. Machine learning and deep learning methods for skin lesion classification and diagnosis: a systematic review. *Diagnostics*, 11(8):1390, 2021.

[19] A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. M. Blei. Automatic differentiation variational inference. *Journal of machine learning research*, 18(14):1–45, 2017.

[20] J. Lee, S. Costello, J. R. Balmes, and S. M. Holm. The association between ambient pm2. 5 and low birth weight in california. *International journal of environmental research and public health*, 19(20): 13554, 2022.

[21] C. Li, D. Gao, Y. S. Cai, J. Liang, Y. Wang, Y. Pan, W. Zhang, F. Zheng, and W. Xie. Relationships of residential distance to major traffic roads with dementia incidence and brain structure measures: mediation role of air pollution. *Health data science*, 3:0091, 2023.

[22] S. Luccioni, Y. Jernite, and E. Strubell. Power hungry processing: Watts driving the cost of ai deployment? In *Proceedings of the 2024 ACM conference on fairness, accountability, and transparency*, pages 85–99, 2024.

[23] G. Marcus. The next decade in ai: four steps towards robust artificial intelligence. *arXiv preprint arXiv:2002.06177*, 2020.

[24] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova, H. Ashrafian, T. Back, M. Chesus, G. S. Corrado, A. Darzi, et al. International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788):89–94, 2020.

[25] G. Menghani. Efficient deep learning: A survey on making deep learning models smaller, faster, and better. *ACM Computing Surveys*, 55(12):1–37, 2023.

[26] National Academies of Sciences, Engineering, and Medicine. *Reproducibility and replicability in science*. National Academies Press, 2019.

[27] S. W. Nijman, A. Leeuwenberg, I. Beekers, I. Verkouter, J. Jacobs, M. Bots, F. Asselbergs, K. G. Moons, and T. P. Debray. Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review. *Journal of clinical epidemiology*, 142:218–229, 2022.

[28] S. Peng, S. Sun, and Y.-D. Yao. A survey of modulation classification using deep learning: Signal representation and data preprocessing. *IEEE Transactions on Neural Networks and Learning Systems*, 33 (12):7020–7038, 2021.

[29] F. Pourpanah, M. Abdar, Y. Luo, X. Zhou, R. Wang, C. P. Lim, X.-Z. Wang, and Q. J. Wu. A review of generalized zero-shot learning methods. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4051–4070, 2022.

[30] J. Rauber, W. Brendel, and M. Bethge. Foolbox: A python toolbox to benchmark the robustness of machine learning models. *arXiv preprint arXiv:1707.04131*, 2017.

[31] S. Scardapane and D. Wang. Randomness in neural networks: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 7(2):e1200, 2017.

[32] J. Snoek, H. Larochelle, and R. P. Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012.

[33] F. Stahlberg. Neural machine translation: A review. *Journal of Artificial Intelligence Research*, 69:343–418, 2020.

[34] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[35] H. Wang, H. Wu, Z. He, L. Huang, and K. W. Church. Progress in machine translation. *Engineering*, 18:143–153, 2022.

[36] W. Wang, V. W. Zheng, H. Yu, and C. Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37, 2019.

[37] L. Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

[38] C.-J. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. Aga, J. Huang, C. Bai, et al. Sustainable ai: Environmental implications, challenges and opportunities. *Proceedings of Machine Learning and Systems*, 4:795–813, 2022.

[39] D. Yogatama and G. Mann. Efficient transfer learning method for automatic hyperparameter tuning. In *Artificial intelligence and statistics*, pages 1077–1085. PMLR, 2014.

[40] T. Yu and H. Zhu. Hyper-parameter optimization: A review of algorithms and applications. *arXiv preprint arXiv:2003.05689*, 2020.

[41] X. Zheng, T. Pang, C. Du, Q. Liu, J. Jiang, and M. Lin. Cheating automatic llm benchmarks: Null models achieve high win rates. *arXiv preprint arXiv:2410.07137*, 2024.