The Bearable Lightness of Big Data: Towards Massive Public Datasets in Scientific Machine Learning

Anonymous Authors[1]

Abstract

In general, large datasets enable deep learning models to perform with good accuracy and generalizability. However, massive high-fidelity simulation datasets (from molecular chemistry, astrophysics, computational fluid dynamics (CFD), *etc.*) can be challenging to curate due to dimensionality and storage constraints. Lossy compression algorithms can help mitigate limitations from storage, as long as the overall data fidelity is preserved. To illustrate this point, we demonstrate that deep learning models, trained and tested on data from a petascale CFD simulation, are robust to errors introduced during lossy compression in a semantic segmentation problem. Our results demonstrate that lossy compression algorithms offer a realistic pathway for exposing high-fidelity scientific data to open-source data repositories for building community datasets. In this paper, we outline, construct, and evaluate the requirements for establishing a big data framework for scientific machine learning.

1. Introduction

Accuracy and generalizability are the requirements of predictive machine learning (ML) models. One way to achieve this is to rely on a wealth of sufficiently high quality data (Sun et al., 2017). In fields such as computer vision, massive and diverse datasets ($\sim$170 Gb, 1.4M images, 1,000 classes) such as ImageNet (Deng et al., 2009), which is shared via Kaggle (Goldbloom & Hamner, 2010), have enabled deep learning models (He et al., 2016) to outperform human capabilities in image recognition (Russakovsky et al., 2015).

In contrast, high-fidelity simulation datasets found in the natural and applied sciences, such as the Johns Hopkins Turbulence Database (Li et al., 2008) are not as diverse (9 simulation cases), but are orders of magnitude greater in size ($\sim$500 TB) due to increased dimensionality and grid resolution requirements. As such, open-source public ML data repositories such as Kaggle (with a size limit of $\mathcal{O}(100)$ GB per dataset) are not feasible. Instead, significant resources and infrastructure must be dedicated towards building and maintaining data storage facilities. Since access to scientific data can be limited, many fields have applied ML in the *small data* regime, where ideas such as physics-informed ML (Karniadakis et al., 2021) are popular.

In fields where high-fidelity simulations are prevalent, the wealth of data required to operate in the *big data* regime does exist. For example, Ihme et al. (2022) identified over 200 high-fidelity simulation cases that could serve as a foundation for a big dataset for turbulent reacting flows. Thus, if these challenges in data storage can be overcome, ML in the natural and applied sciences could more effectively leverage advances from the broader ML community – which focuses on big data, big models (Yuan et al., 2022), and foundation models (Bommasani et al., 2021) – towards predictive tasks in scientific problems. We must note that both small and big data paradigms do not necessarily compete, and that effective data-driven models, in fields such as material science (Lu et al., 2020), flow physics (Wu et al., 2018), and astro-physics (Khan et al., 2020), have been developed by combining ideas from both paradigms.

Here, we propose lossy compression methods (Cappello et al., 2019) towards compressing data into tractable sizes, suitable for sharing via public repositories, at the cost of introducing errors (typically controllable via error-bounded algorithms) to a dataset. This is complemented by a recent study (Northcutt et al., 2021) which demonstrated that ImageNet and other popular benchmark datasets contain up to 10% label errors. Despite these errors in training data, ML continues to perform with remarkable accuracy because modern deep learning algorithms are inherently robust to noisy data (Rolnick et al., 2017; Mahajan et al., 2018). This means that lossy compression could be applied towards mitigating storage limits in public repositories.

In this work, we propose a realistic pathway for building high-fidelity scientific datasets required to operate in the

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

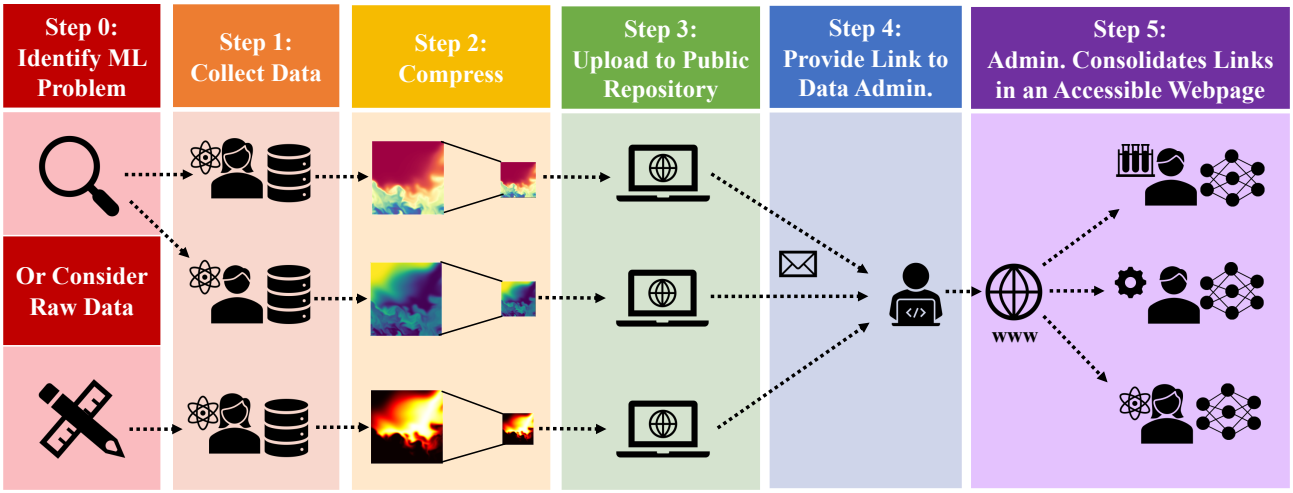


Figure 1. A path towards public datasets in scientific ML.

big data regime. This framework – combining lossy compression, community outreach, and public repositories – is summarized in Figure 1. A preliminary step before data collection involves either (i) identifying target supervised learning problems, where labels can be defined, or (ii) choosing to simply share raw scientific data. Next, the data is collected and compressed into a consistent data format with at a desired level of error. Then, the compressed dataset ($\mathcal{O}(100)$ GB each) from different scientific investigators can be uploaded onto a public ML repository. A link and description of the dataset can then be shared to a data administrator, who curates the links and metadata from the network of distributed datasets on a community-hosted webpage.

This big scientific data framework relies on the (i) the size and quality of the compressed data, and (ii) the robustness of deep learning models to errors introduced during lossy compression. To address these concerns, we quantify the errors and reduction offered by a lossy compression algorithm, SZ2 (Liang et al., 2018b), and demonstrate that deep learning is still effective after training on lossy data extracted from a petascale turbulent reacting flow simulation, in a semantic segmentation problem. Within turbulent reacting flows, this type of classification can be useful for detecting catastrophic rare events (Cellier et al., 2021), optimizing numerical computations (Chung et al., 2021), and identifying combustion regimes (Wan et al., 2020). In other scientific fields, semantic segmentation have been explored for processing data extracted from microscopes (Ronneberger et al., 2015), radio-astronomical measurements (Pino et al., 2021), and neutrino experiments (Abratenko et al., 2021). The present data is further described in Section 2, while the lossy compression algorithm and the 3-D convolutional neural network (CNN) employed are detailed in Section 3. We present our results and conclusions in Section 4 and Section 5, respectively.

2. Data Description

A three-dimensional direct numerical simulation (DNS) dataset for a turbulent lifted hydrogen jet flame in heated coflow air (Jung et al., 2021) is used in this study. This simulation data was generated by solving the compressible Navier-Stokes equations, along with species continuity and total energy conservation equations, using a high-order numerical solver (Chen et al., 2009), with a detailed 9-species and 21-reaction hydrogen-air chemical mechanism (Li et al., 2004).

This petascale simulation of reacting flow in a slot-burner configuration consists of 1.28 billion grid points ($2000 \times 1600 \times 400$) with 12 conserved quantities for the flow-fields. A single snapshot of this data is slightly more than 100 GB in size. A diluted hydrogen fuel is issued from the central jet with a jet Reynolds number of 8,000. The central jet is surrounded on either side by co-flowing heated air streams with an inlet temperature of 850 K. The computational domain size is $30 \times 40 \times 6$ mm^3 in the streamwise, x-, transverse, y-, and spanwise, z- directions, respectively. A uniform grid size of 15 μm is placed in x- and z- directions while and algebraically-stretched grid is adopted in the y- directions.

In the present study, a sub-region (with 60M cells) of the DNS field (i.e., a left half branch of the lifted jet flame) is sampled to evaluate the lossy compression algorithm. From this data, we extract four flow features, and generate five classes of labels from this dataset, and subdivide the data into 268 sub-volumes, each with $256 \times 256 \times 3$ cells and four channels for the features in the input flow-field. Note

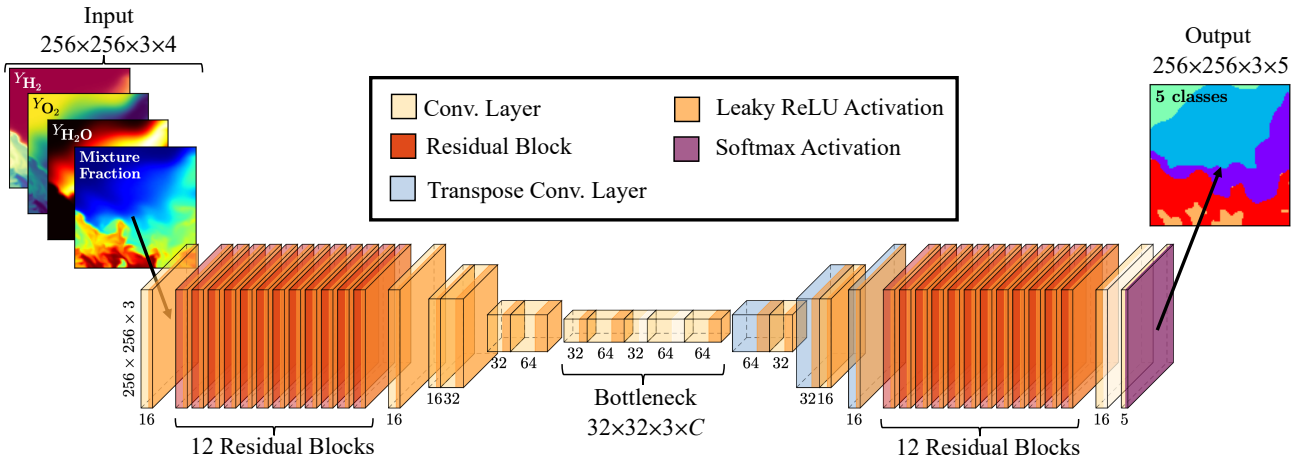


Figure 2. Present 3-D CNN architecture. Number of filters per layer C are described at the bottom of each layer.

that since this configuration is homogeneous in the spanwise direction, 3 cells in the z-axis is sufficient for preserving spatial information in these subvolumes. The features consist of mass fractions of major chemical species and mixture fraction[1] as defined by Bilger (1976), i.e. $\{Y_{H_2}, Y_{O_2}, Y_{H_2O}, Z\}$, and are normalized with a min-max scaler prior to training. The classes consist of premixed flame, non-premixed flame, pure fuel, pure air, and unburned fuel-air mixture which can be generated directly from the magnitudes and gradients (Yamashita et al., 1996) of the input features. Train, validation, and test sets are split in a typical 60:20:20 fashion, with random rotation and random flipping used to further augment the train set.

3. Methods

3.1. Deep Learning Model

The architecture of the 3-D convolutional neural network (CNN) employed is also shown in Figure 2. This architecture is based on the work of Glaws et al. (2020), and is known to perform effectively in flow physics problems, with the input size of the present model modified to $256 \times 256 \times 3 \times 4$, the minimum number of filters per hidden layer increased to 16, and the filter width reduced to 3. 12 residual blocks are placed before and after an autoencoder network, with a softmax output activation for 5 classes used together with a categorical cross-entropy loss function to solve the present semantic segmentation problem. This network contains 93 layers and approximately 1M trainable parameters, with weights initialized via Xavier initialization (Glorot & Bengio, 2010). Train and validation procedures are shared in Appendix A.

[1]The mixture fraction can be thought of as reduced dimension of chemical composition of a reacting fuel-air mixture, with values of 0 and 1 corresponding to the amount of material originating from the oxidizer and fuel streams, respectively.

3.2. Lossy Compression Algorithm

We employ the SZ2 compressor (Liang et al., 2018b), which combines curve-fitting, the Lorenzo predictor, and data quantization, tailored to compressing a wide range of scientific data including measurements from seismic imaging and X-ray, as well as simulation data in molecular dynamics, cosmology, and flow physics. In principle, this compression algorithm (i) partitions field variables into neighborhoods, (ii) iteratively searches for approximate regression functions that can describe each neighborhood with a guaranteed error-bound, and (iii) stores the quantized regression coefficients of the function and indices of the field variables for reconstructing the data during decompression. Since the quantized coefficients and indices are much smaller than the original field variables, the data can be compressed more effectively than lossless compression algorithms.

For this study, we consider the point-wise relative error method (Liang et al., 2018a) within this compressor, which guarantees that the lossy error in each cell does not exceed a user-defined percentage of the compressed value. This method results in a well-defined measure of quality for describing any shared scientific datasets in a public repository, and is especially useful for maintaining the fidelity of lossy compressed data with large variances, such as with flow velocity and mass fractions of minor chemical species such as OH.

4. Results

4.1. Effects of Lossy Compression on Data

We first compress 3-D scalar fields from the entire learning set with SZ2. Figure 3 demonstrates the total compression

ratio[2] from 1% to 50% max point-wise error ranges from 6- to 23-fold compression. Even if we consider only the smallest compression ratio seen in compressing Y_{H_2O}, a 5-fold compression of the raw ∼100 GB petascale simulation dataset, would enable at least 4 snapshots of this data to be shared as a single dataset on Kaggle. Data compression could be repeated on other flow configurations, and shared via the framework presented in Figure 1 for building a distributed ML training dataset.

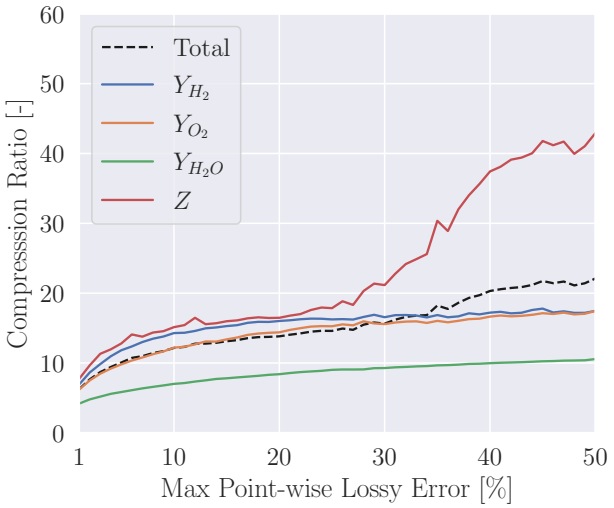


Figure 3. SZ2 (Liang et al., 2018b) compression ratio of features of the entire learning set at different specified maximum point-wise error.

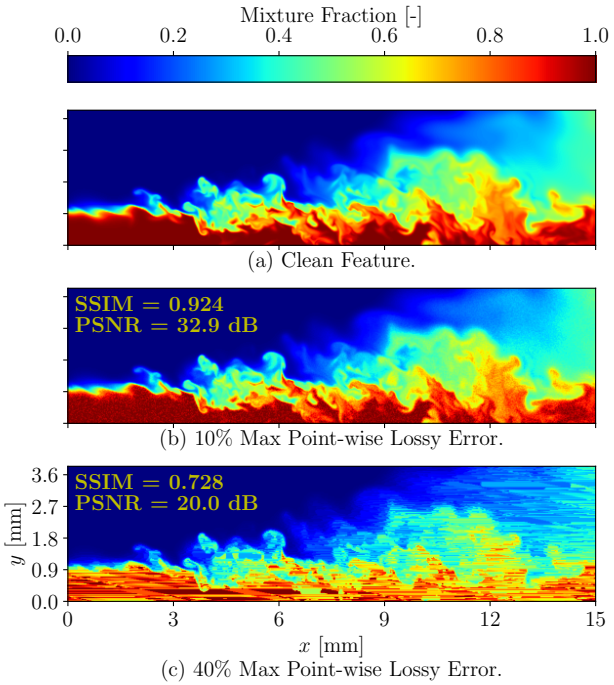
Next, we decompress the compressed data and evaluate errors introduced by the lossy compressor. Figure 4 compares mixture fraction Z at different levels of maximum point-wise lossy error, with the original clean feature. Image quality metrics (Horé & Ziou, 2010) such as peak-signal-to-noise-ratio[3] (PSNR) and structural similarity index measure[4] (SSIM) are shown to decrease with increasing compression, with large field distortions observed with 40% max point-wise error (Figure 4c).

After decompressing the scalar fields, we generate the training labels for the different lossy data, as described in Section 2. Figure 5 compares the five classes at different levels of maximum point-wise error, with the original clean training label. At Figure 5b, significant noise is seen especially in the premixed and non-premixed flame regions at 10% max point-wise lossy error, with a 9.3% total label error

[2]Compression ratio is defined as the original file size divided by the compressed file size.

[3]Higher PSNR means higher image quality. Note that the highest possible value for PSNR is 48 dB for 8-bit integers, and 760 dB for single-precision floating points.

[4]Higher SSIM means higher image quality. SSIM is bounded between -1 and +1.



(a) Clean Feature.

(b) 10% Max Point-wise Lossy Error.

(c) 40% Max Point-wise Lossy Error.

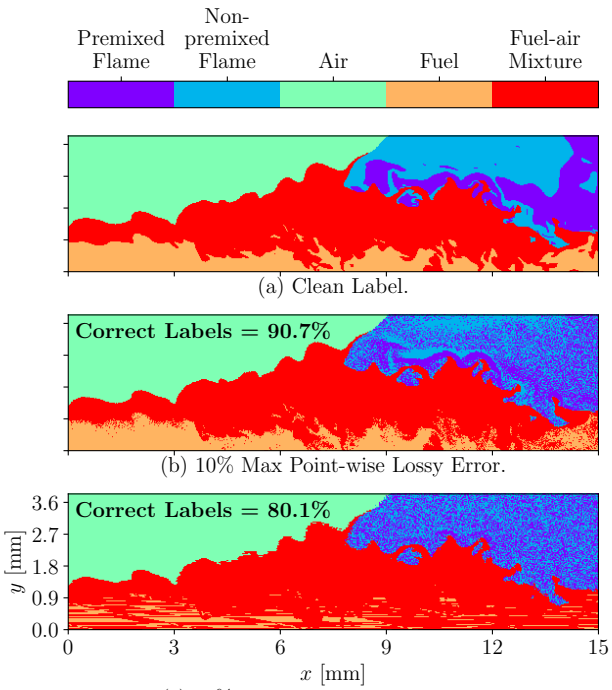
Figure 4. A feature (mixture fraction) from the train set at different levels of maximum pointwise-error specified during compression. Quality metrics such as peak-signal-to-noise-ratio (PSNR) and structural similarity index measure (SSIM) are included.

introduced to the data. This noise is present because scalar gradients, used in generating the flame labels (Yamashita et al., 1996), are not necessarily preserved adequately after lossy compression. Figure 5b shows that the fuel labels become especially distorted at 40% max point-wise lossy error, with a total label error 19.9%.

4.2. Train with Lossy data, and Test on Clean Data

In general, validation and test sets do not come from the same distribution as the training data, and are usually sampled to represent data encountered after deployment. Thus, in the big data framework proposed in Figure 1, we envision a scenario where large quantities of lossy compressed training data can be easily obtained from public repositories, with small quantities of clean test and validation data sampled personally by a user.

In this study, we explore effects of training with lossy data, and testing and validating on clean data. This task can be further subdivided into two scenarios: (i) where lossy features are shared into the repository with clean labels, and (ii) lossy labels are generated from lossy data obtained from the repository (such as with Figure 5). The former scenario is encountered where a specific target supervised learning problem (such as with ImageNet for image recognition) has been identified. In this case, lossy compression does not necessarily need to be employed to the labels, since



(a) Clean Label.

(b) 10% Max Point-wise Lossy Error.

(c) 40% Max Point-wise Lossy Error.

Figure 5. Training labels generated from lossy data at different levels of maximum pointwise-error specified during compression.

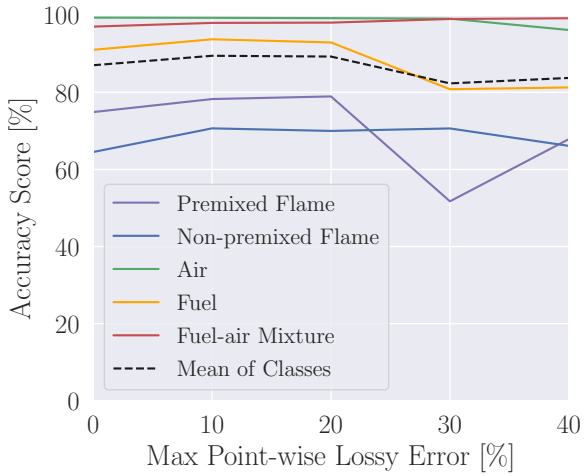
the dimensionality of labels are much smaller than features. This could be more beneficial than extracting potentially noisy labels from lossy scientific data in the latter scenario, especially since ML methods are well-known to be more robust to feature noise than label noise (Zhu & Wu, 2004).

4.2.1. Lossy Features and Clean Labels

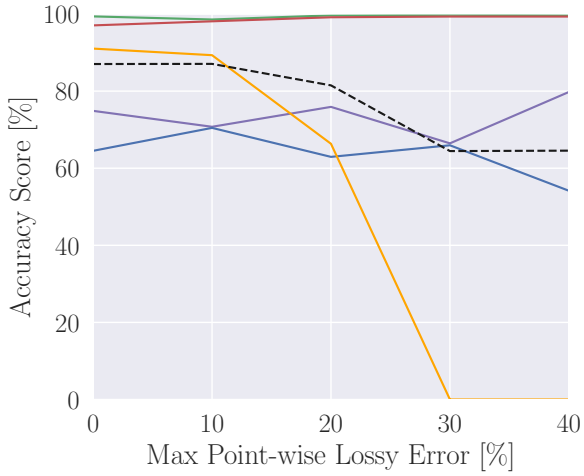
Figure 6a compares class-specific accuracy scores, along with the mean of these scores, for different levels of maximum point-wise lossy errors, when training with lossy features and clean labels. Mean accuracy score of 80% is seen in the baseline case of 0% lossy error, which is typical in a semantic segmentation problem (Ronneberger et al., 2015). The mean accuracy scores are seen to be robust up to 20% max point-wise lossy error, which corresponds to a 14-fold compression in the original data.

4.2.2. Lossy Features and Lossy Labels

Figure 6b compares class-specific accuracy scores, along with the mean of these scores, for different levels of maximum point-wise lossy errors,when training with both lossy features and lossy labels. The mean accuracy scores are seen to be robust to errors in the features up to only 10% max point-wise lossy error, which still corresponds to a 12-fold compression in the original data. After 20% max lossy error, class accuracy for fuel drops promptly to 0, which



(a) Model trained on lossy features and clean labels, and tested on clean and uncompressed features.



(b) Model trained on lossy features and lossy labels, and tested on clean and uncompressed features.

Figure 6. Class accuracy score at different levels of maximum pointwise-error specified during compression.

can be explained by the highly distorted training labels for fuel shown in Figure 5c. Remarkably, the deep learning model demonstrates reasonably robust behavior in the other classes, especially in the flame regions, to max point-wise lossy errors up until 40% .

Figure 7 visualizes the predictions from the deep learning model trained on lossy features and lossy labels, and tested on clean features and labels. Figure 7b shows that the model predictions at 0% lossy error are in reasonable agreement with the ground truth labels in Figure 7a. Evident misclassification of non-premixed flame is seen near the boundary with air in Figure 7b and Figure 7c. This is likely caused by the presence of noisy labels between the premixed and non-premixed flame labels seen in Figure 5. Nevertheless, coherent classification is still observed in the flame regions,

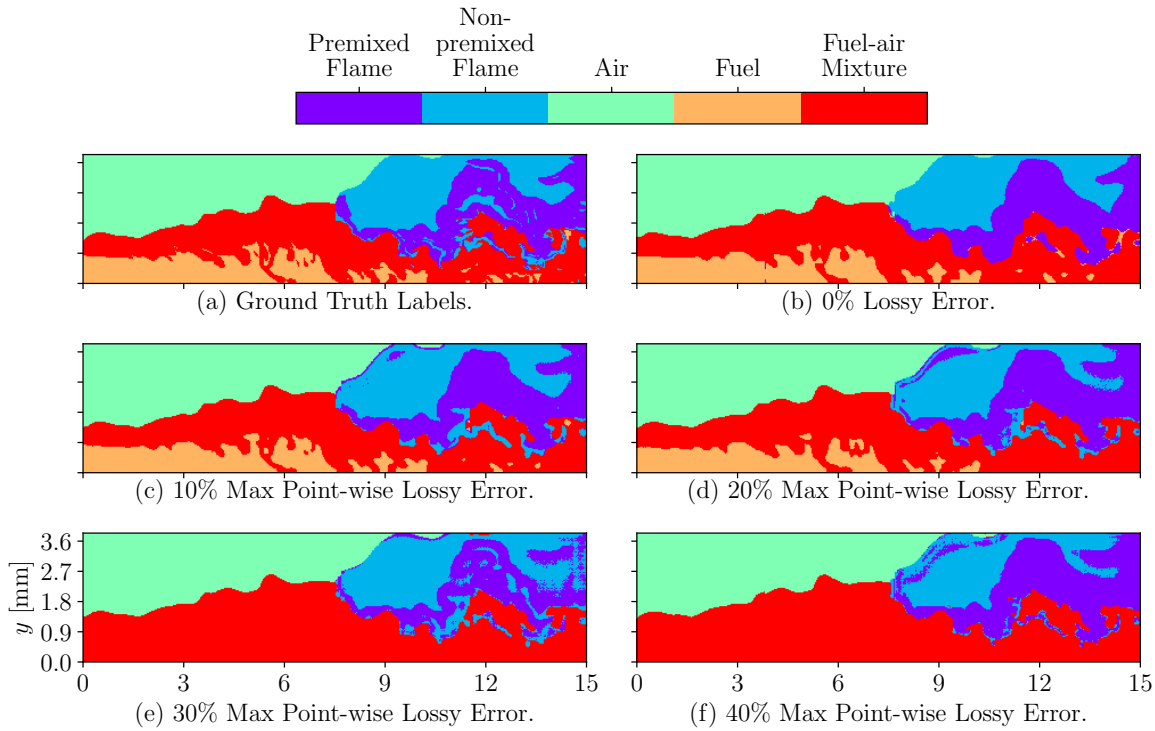


Figure 7. Visualization of ground truth and predictions from model trained on lossy features and lossy labels, and tested on clean features and labels.

despite the noisy training labels up to 40% max point-wise lossy error in Figure 5c, as previously discussed with the class accuracy scores in Figure 6b.

5. Conclusions

In this paper, we provide the requirements for establishing a realistic big data framework for scientific ML. These requirements are (i) community involvement, (ii) public data repositories, and (iii) lossy compression algorithms.

To demonstrate that lossy data is useful for training ML algorithms, we compress data from a petascale simulation of a turbulent reacting flow configuration, and employ the compresssed data to train a 3-D CNN in a semantic segmentation problem. Two scenarios are investigated: (i) where lossy features are shared into the repository with clean labels, and (ii) where lossy labels are generated from raw lossy data obtained from the repository. In the case of only lossy features, the CNN is robust up until 20% max point-wise lossy error, corresponding to a compression of 14-fold. The CNN is robust up until 10% max point-wise lossy error, corresponding to a 12-fold compression ratio. These results indicate that accurate predictions can still be made by deep learning algorithms even when training with lossy data, and that lossy compression can be utilized to mitigate storage constraints in open-source data repositories.

We intend to extend the analysis presented here to consider more complex regression problems through future studies. Nevertheless, the results from this work show that with community involvement, public data repositories, and lossy compression algorithms, the challenging task of creating and storing big data for scientific ML can be more bearable.

Supplementary Material

The code, models, and data employed in this study will be made published through GitHub and Kaggle. In addition, the links to proof-of-concept website proposed in Figure 1 will be included here, following the double-blind review process.

References

Abratenko, P., Alrashed, M., An, R., Anthony, J., Asaadi, J., Ashkenazi, A., Balasubramanian, S., Baller, B., Barnes, C., Barr, G., Basque, V., et al. Semantic segmentation with a sparse convolutional neural network for event reconstruction in MicroBooNE. *Physical Review D*, 103: 052012, 2021.

Bilger, R. Turbulent jet diffusion flames. *Progress in Energy and Combustion Science*, 1(2):87–109, 1976.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R.,

Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv pre-print 2108.07258*, 2021.

Cappello, F., Di, S., Li, S., Liang, X., Gok, A. M., Tao, D., Yoon, C. H., Wu, X.-C., Alexeev, Y., and Chong, F. T. Use cases of lossy compression for floating-point data in scientific data sets. *The International Journal of High Performance Computing Applications*, 33(6):1201–1220, 2019.

Cellier, A., Lapeyre, C., Öztarlik, G., Poinsot, T., Schuller, T., and Selle, L. Detection of precursors of combustion instability using convolutional recurrent neural networks. *Combustion and Flame*, 233:111558, 2021.

Chen, J. H., Choudhary, A., de Supinski, B., DeVries, M., Hawkes, E. R., Klasky, S., Liao, W. K., Ma, K. L., Mellor-Crummey, J., Podhorszki, N., Sankaran, R., Shende, S., and Yoo, C. S. Terascale direct numerical simulations of turbulent combustion using S3D. *Computational Science and Discovery*, 2(1):015001, 2009.

Chung, W. T., Mishra, A. A., Perakis, N., and Ihme, M. Data-assisted combustion simulations with dynamic submodel assignment using random forests. *Combustion and Flame*, 227:172–185, 2021.

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 the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 248–255, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

Glaws, A., King, R., and Sprague, M. Deep learning for in situ data compression of large turbulent flow simulations. *Physical Review Fluids*, 5:114602, 2020.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 2010 International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, pp. 249–256, 2010.

Goldbloom, A. and Hamner, B. Kaggle: Your machine learning and data science community, 2010. https://www.kaggle.com.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, pp. 770–778, Los Alamitos, CA, USA, 2016. IEEE Computer Society.

Horé, A. and Ziou, D. Image quality metrics: PSNR vs. SSIM. In *Proceedings of the 2010 IEEE International Conference on Pattern Recognition (ICPR 2010)*, pp. 2366–2369, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

Ihme, M., Chung, W. T., and Mishra, A. A. Combustion machine learning: Principles, progress and prospects. *Progress in Energy and Combustion Science*, 91:101010, 2022.

Jung, K. S., Kim, S. O., Lu, T., Chen, J. H., and Yoo, C. S. On the flame stabilization of turbulent lifted hydrogen jet flames in heated coflows near the autoignition limit: A comparative DNS study. *Combustion and Flame*, 233: 111584, 2021.

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

Khan, A., Huerta, E., and Das, A. Physics-inspired deep learning to characterize the signal manifold of quasi-circular, spinning, non-precessing binary black hole mergers. *Physics Letters B*, 808:135628, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv pre-print 1412.6980*, 2014.

Li, J., Zhao, Z., Kazakov, A., and Dryer, F. L. An updated comprehensive kinetic model of hydrogen combustion. *International Journal of Chemical Kinetics*, 36(10):566–575, 2004.

Li, Y., Perlman, E., Wan, M., Yang, Y., Meneveau, C., Burns, R., Chen, S., Szalay, A., and Eyink, G. A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, (9):N31, 2008.

Liang, X., Di, S., Tao, D., Chen, Z., and Cappello, F. An efficient transformation scheme for lossy data compression with point-wise relative error bound. In *Proceedings of the 2018 IEEE International Conference on Cluster Computing (CLUSTER 2018)*, pp. 179–189, Los Alamitos, CA, USA, 2018a. IEEE Computer Society.

Liang, X., Di, S., Tao, D., Li, S., Li, S., Guo, H., Chen, Z., and Cappello, F. Error-controlled lossy compression optimized for high compression ratios of scientific datasets. In *Proceeding of the 2018 IEEE International Conference on Big Data (Big Data 2018)*, pp. 438–447, Los Alamitos, CA, USA, 2018b. IEEE Computer Society.

Lu, L., Dao, M., Kumar, P., Ramamurty, U., Karniadakis, G. E., and Suresh, S. Extraction of mechanical properties of materials through deep learning from instrumented indentation. *Proceedings of the National Academy of Sciences*, 117(13):7052–7062, 2020.

Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., and van der Maaten, L. Exploring the limits of weakly supervised pretraining. In *Proceedings of the 2018 European Conference on Computer Vision (ECCV 2018)*, pp. 185–201, Cham, Switzerland, 2018. Springer International Publishing.

Northcutt, C., Athalye, A., and Mueller, J. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Proceedings of the 2021 Neural Information Processing Systems Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks 2021)*, 2021.

Pino, C., Sortino, R., Sciacca, E., Riggi, S., and Spampinato, C. Semantic segmentation of radio-astronomical images. In Hernández Heredia, Y., Milián Núñez, V., and Ruiz Shulcloper, J. (eds.), *Proceedings of the 2021 International Workshop on Artificial Intelligence and Pattern Recognition (IWAIPR 2021)*, pp. 393–403, Cham, Switzerland, 2021. Springer International Publishing.

Rolnick, D., Veit, A., Belongie, S. J., and Shavit, N. Deep learning is robust to massive label noise. *arXiv pre-print 1705.10694*, 2017.

Ronneberger, O., Fischer, P., and Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F. (eds.), *Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015)*, pp. 234–241, Cham, Switzerland, 2015. Springer International Publishing.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3): 211–252, 2015.

Sun, C., Shrivastava, A., Singh, S., and Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV 2017)*, pp. 843–852, Los Alamitos, CA, USA, 2017. IEEE Computer Society.

Wan, K., Hartl, S., Vervisch, L., Domingo, P., Barlow, R. S., and Hasse, C. Combustion regime identification from machine learning trained by Raman/Rayleigh line measurements. *Combustion and Flame*, 219:268–274, 2020.

Wu, J.-L., Xiao, H., and Paterson, E. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids*, 3(7):074602, 2018.

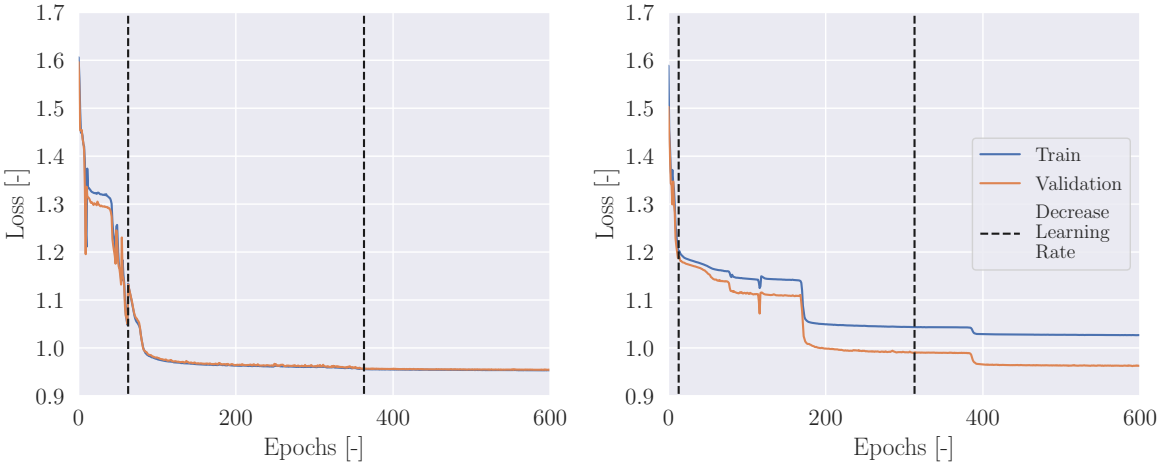
Yamashita, H., Shimada, M., and Takeno, T. A numerical study on flame stability at the transition point of jet diffusion flames. *Proceedings of the Combustion Institute*, 26 (1):27–34, 1996.

Yuan, S., Zhao, H., Zhao, S., Leng, J., Liang, Y., Wang, X., Yu, J., Lv, X., Shao, Z., He, J., et al. A roadmap for big model. *arXiv pre-print 2203.14101*, 2022.

Zhu, X. and Wu, X. Class noise vs. attribute noise: A quantitative study. *Artificial Intelligence Review*, 22(3): 177–210, 2004.

A. Training and Validation

Training is performed with the Adam (Kingma & Ba, 2014) optimizer, with a batch size of 24 and raw learning rates of 1E-4, 1E-5, and 1E-6 for 100, 300, and 300 epochs, respectively, with early-stopping employed when necessary. Prior to training, the raw learning rates are multiplied by the square root of the batch size. Note that in Figure 8b, the converged validation loss can be lower than the training loss, leading to higher validation accuracy than training accuracy. This is caused by the absence of lossy errors in the validation set, as described in Section 4. Training this model on four Tesla V100 GPUs requires a total of approximately 4 hours of wall-clock-time for each case.



(a) 20% max point-wise lossy error in only features.

(b) 10% max point-wise lossy error in both features and labels.

Figure 8. Loss during training.