Training Data Error Impacts on Deep Neural Networks for Classifying RNA-seq Expressions

Rajeev Jain,* Jamaludin Mohd-Yusof,† Justin M. Wozniak,‡ Fangfang Xia,§ Thomas Brettin,§ Rick Stevens§
* Mathematics and Computer Science, Argonne National Laboratory, Lemont, IL, USA
† Computer, Computational and Statistical Sciences, Los Alamos National Laboratory, Los Alamos, NM, USA
‡ Data Science and Learning, Argonne National Laboratory, Lemont, IL, USA
§ Computing, Environment, and Life Sciences, Argonne National Laboratory, Lemont, IL, USA

Advanced analyses and computations based on gene expressions are prone to errors as they depend on experimental design, chemical operations/measurements and data analysis. The assembly and aggregation of such data for creating deep neural network models may further influence the accuracy of these analyses. For example, the CANDLE [1] NT3 Benchmark [2] used in this paper attempts to separate tumor tissue from normal tissue using gene-expression-level sample signatures. The associated deep neural network (DNN) has an input layer for RNA sequence gene expression. It is a 1D convolutional network for classifying RNA-seq gene expression profiles into normal or tumor tissue categories. The network follows the classic architecture of convolutional models with multiple 1D convolutional layers interleaved with pooling layers followed by final dense layers. The network can optionally use 1D locally connected layers in place of convolution layers as well as dropout layers for regularization.

A. Model overview

The model is trained/tested on the matched normal-tumor gene expression profile pairs available from the NCI genomic data commons [3] respectively. The full set of expression features contains 60,483 float columns transformed from RNA-seq FPKM-UQ [4] values. Before our modifications, this model achieved around 98% classification accuracy. The benchmark runs in this paper used the CANDLE hyperparameters shown in Table I. In this work, we use the NT3 Benchmark to study the effects of injecting bad data at different rates to study the impacts on the resulting predictions. Our data manipulations include flipping classification labels (label noise) and introducing noise in gene expressions (feature noise). Our experiments used various trial sizes and injection rates for each model to understand the effects on the accuracy of the neural network. We ran large ensembles of these training runs on OLCF *Summit* to observe the error impacts over a large range and with multiple trials.

Network architecture		**Training limits**	
conv	[(64, 20, 1), (64, 10, 1)]	epochs	100
pool	[1, 10]	timeout	3600.0
dense	[200, 20]	**Noise injection**	
classes	2	noise_add	true
out_act	'softmax'	noise_gaussian	false
activation	'relu'	noise_level	0.2
Training settings		noise_correlated	true
optimizer	'sgd'	noise_labels	0.2
loss	'categorical_crossentropy'	feature_threshold	0.01
metrics	'accuracy'	feature_col	11180
batch_size	16		
learning_rate	0.002		
drop	0.0		

TABLE I
HYPERPARAMETERS USED FOR NT3 BENCHMARK IN THIS PAPER.

We introduce noise into the data in two ways, on the labels and on the features. For the labels, we consider both uncorrelated and correlated noise, set by **noise_correlated** . For uncorrelated noise, we randomly flip the normal/tumor labels on a fraction of the samples corresponding to the **noise_level**. In correlated label noise, we perform the same type of label flips, but only on samples where the expression on a certain gene (set by **feature_col**) is above a certain threshold (**feature_threshold**), so that the prevalence of noisy labels is correlated with the expression of that gene.

For correlated label noise, there are some factors to consider. The first is the number of samples in which the gene is expressed; if the gene is rarely above the threshold, then the total number of sample eligible for noise injection is small, and the effective noise added will likewise be small. Second, we must consider whether the gene is itself correlated with the normal/tumor label value; if not, then injection of label noise correlated with that gene is unlikely to confuse the training process. In this study we choose to inject correlated label noise on a feature which is highly correlated with the labels and sufficiently prevalent in the samples.

For feature noise, we again consider two types of noise injection: increasing the feature values by a fixed percentage, and introducing Gaussian random noise across the features.

B. Label noise

1) Correlated label noise: In Figure 1 we compare the validation performance of the abstaining model with the base model. The abstaining model retains much higher accuracy than the base model on the validation set, but must abstain on a higher fraction of the samples than the naive $1 - BaseAcc$ estimate. In fact by 50% noise level, the abstaining classifier has learned that the labels are no longer predictive and abstains on virtually all of the data.

2) Uncorrelated label noise: The Figure 2 shows that the behavior is broadly similar to that for uncorrelated noise; by 50% noise injection, the abstaining classifier has learned

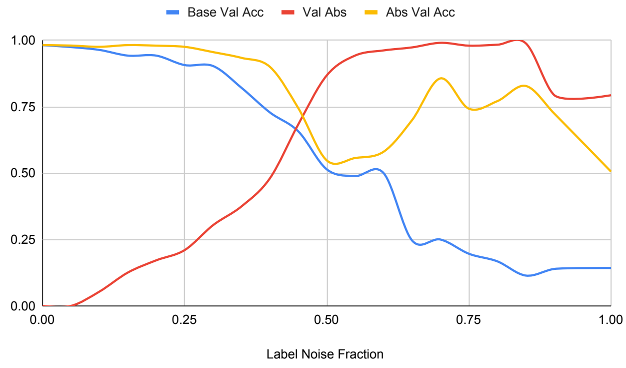

Fig. 1. Comparison of validation accuracy and abstention with the base model in the presence of uncorrelated label noise.

that the labels are unreliable and abstains on virtually all the samples, while formally retaining accuracy on the (small) fraction of samples it attempts to classify. While we expected more divergence between the uncorrelated and correlated noise scenarios, this may be obscured by the fact that this problem is binary classification and the limited number of classes.

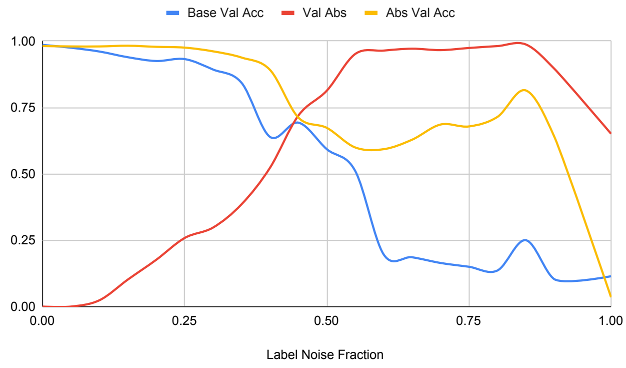

Fig. 2. Comparison of validation accuracy and abstention with the base model in the presence of correlated label noise.

C. Feature noise

1) Constant percentage feature noise injection: In another round of tests, for each error percentage r, all the 60,483 RNA sequence gene expression are increased by a fixed percentage value $r \times z$, where z is a random number in $[0, 1)$. The results indicate that this noise injection does not have any effect on the accuracy of the model. This may be due to the fact that the model gets scaled by the same factor as the percentage bad data injection, and many of the RNA values are 0 (zero). This indicates that the model is not significantly affected by scaling errors in the training data. For this reason we do not use the abstaining classifier on this dataset.

2) Gaussian feature noise injection : In Figure 3 we compare the performance of the base and abstention models. Although the base model is able to retain good accuracy on the validation set, the abstention model nevertheless improves on this performance, while again abstaining on a fraction of the data approximately twice that of the naive estimate. This is consistent with the notion that with only two classes, the base model can perform better by a factor of two due to "lucky guesses."

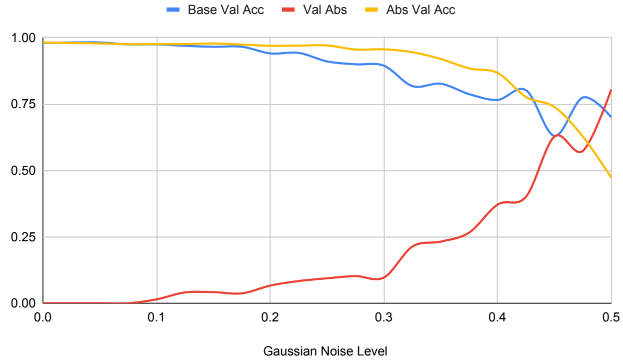

Fig. 3. Comparison of validation accuracy and abstention with the base model in the presence of Gaussian feature noise.

D. Conclusion

The introduction of either correlated or uncorrelated label noise results in increased validation loss and reduced validation accuracy in the base model. With the introduction of a constant percentage noise in gene expression and label noise the validation accuracy and validation loss remains mostly constant. In contrast, when introducing random Gaussian noise with varying standard deviation to gene expressions for all the samples we see validation accuracy declines sharply indicating a poor fit compared to model with the original data. We introduce an abstaining [5] version of the model, which adds an extra abstention class, allowing the model to abstain when it is not confident of the prediction. This model retains accuracy while abstaining on progressively higher fractions of the data as more noise is injected.

The contributions of this work include 1) a description of the noise impacts on the well-known cancer benchmark NT3, 2) a description of a recently developed classifier to handle uncertain predictions, and 3) experimental results from the application of the classifier to the problem of training a cancer benchmark with noisy data. The results confirm the robustness of the input data and the deep neural network model represented by the CANDLE NT3 benchmark, especially when used in conjunction with the abstention approach. Such noise analysis studies would help set error tolerance for actual experimental measurement of RNA-seq gene expression profiles.

E. Future Work

Additional error studies motivated by real-world error experimental error modalities are needed to develop confidence in deep learning-driven analysis of lab data. Careful segregation of gene expression along specific error prone regions can also be studied to identify bad data or discover interesting data. Other cancer study data could also be subject to relatively simple error analysis, aided by the large compute power that can now be applied to training cancer analysis models.

References

[1] J. M. Wozniak, R. Jain, P. Balaprakash, J. Ozik, N. Collier, J. Bauer, F. Xia, T. Brettin, R. Stevens, J. Mohd-Yusof, C. G. Cardona, B. V. Essen, and M. Baughman, "CANDLE/Supervisor: A workflow framework for machine learning applied to cancer research," *BMC Bioinformatics*, vol. 19, no. 18, p. 491, 2018. [Online]. Available: https://doi.org/10.1186/s12859-018-2508-4

[2] "CANDLE NT3 GitHub." [Online]. Available: https://github.com/ECP-CANDLE/Benchmarks/tree/master/Pilot1/NT3

[3] "NCI web site," https://gdc.cancer.gov/.

[4] "GDC Documentation HTSeq-FPKM-UQ web site," https://docs.gdc.cancer.gov/Encyclopedia/pages/HTSeq-FPKM-UQ/.

[5] S. Thulasidasan, T. Bhattacharya, J. Bilmes, G. Chennupati, and J. Mohd-Yusof, "Combating label noise in deep learning using abstention," in *Proceedings of the 36-th International Conference on Machine Learning*, 2019.

Acknowledgments

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under contract number DE-AC02-06CH11357. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.