Using gradients to explain deep neural network behavior against out-of-distribution data

CSCI 6917, Guided Research
Project proposal
Jafar Isbarov

1. What are you going to do?

Gradients are one of the main characteristics that are used to explain the behavior of deep neural networks (DNNs). There have been some attempts to utilize gradient values as a measure of uncertainty in DNNs. However, no work has attempted to fully explain how gradients change when the DNN is introduced to an out-of-distribution (OOD) data point. I am going to attempt this with the following steps:

(1) Develop a Python package that can be used to log, visualize, and analyze the network gradients during training. This package should be compatible with PyTorch models. (TensorBoard offers such features but in a limited fashion.)
(2) Perform OOD training experiments with multiple models and datasets to collect the necessary data.
(3) Perform qualitative and quantitative analysis of these experimental results to explain how gradients behave w.r.t. OOD data. More specifically:
    (a) Analyze how these gradient differences correlate with model performance (accuracy, confidence scores, and error rates on ID and OOD data).
    (b) Investigate the effects of regularization on gradient behavior.

2. How is it done today? Current Limitations?

[1] tracks weight trajectories and uses this information to estimate model uncertainty. They provide the source code, but it is not a package, hence not easily reusable. I intend to provide an open-source package for this.

[2], [3] and [5] use backpropagated gradients for uncertainty estimation. While this method does not use historical data, it still relies on training metadata to achieve results.

[4] starts with the same ideas, but they provide a more in-depth explanation for gradient behavior by identifying two different types of abnormality: the zero-deflation abnormality and the channel-wise average abnormality.

While the gradients are utilized, none of these works concentrate on the behavior of gradient. I am not aware of any other works that do this.

Another limitation is the lack of an easily usable library integrated with a major deep learning framework. PyTorch supports adding hooks to its neural network modules. It is a good starting point for developing the software. Our toolkit will support all PyTorch models out of the box.

3. What is your new idea?

A systemic review of the gradient behavior w.r.t. OOD data. Instead of trying to improve the performance of uncertainty estimation methods, I will concentrate on the different behavior w.r.t. in-distribution and out-of-distribution data. We expect new insights regarding:

- Relationship between model uncertainty and gradients
- Gradient behavior against adversarial attacks
- OOD effects on mini-batch gradients (more variance than ID?)
- **In which layers do we observe more difference in gradients of OOD and ID data?**

4. Who will benefit from your work? Why?

Researchers who are trying to gain insight into how a deep neural network learns the data distribution. This project can also help us develop a more intuitive understanding of the training process. It will also help us better understand and potentially improve uncertainty estimation methods that rely on gradients.

5. What risks do you anticipate?

We expect little technical challenge since we will be building on top of existing frameworks like PyTorch. One potential issue is that we will not be able to support all PyTorch models out-of-box. If this is the case, the toolkit can be expanded later to include more models.

It is of course possible that our work will not reveal interesting insights, since it is an explorative work.

6. Out-of-pocket costs? Complete within 11 weeks?

We can have an expense of around 100 USD for GPU usage. The project can be finished in less than 11 weeks. If successful, it can be extended to be a Master's thesis.

7. Midterm Demonstration

This project has two parts: developing the necessary tools and building up the theory. I intend to do these in multiple iterations rather than finishing one before starting the other. The midterm demonstration will contain the following:

- A more comprehensive literature review on (1) uncertainty estimation and (2) any work that analyzes gradient behavior.
- A Python package that will allow researchers to track gradients through the training process.
- Visualization and basic analytics using this package on simple deep learning models.

8. Final Demonstration

The final demonstration will contain the following:

- The entire toolkit including the codebase and open-source development guidelines.
- More advanced samples with RNN, LSTM, and possibly attention mechanism.
- A final report on the behavioral differences against in-distribution and out-of-distribution data, potential application areas, and future research directions.

Some changes are possible depending on how the project progresses and what new research we discover.

REFERENCES

[1] Franchi, G., Bursuc, A., Aldea, E., Dubuisson, S., & Bloch, I. (2019). TRADI: Tracking deep neural network weight distributions. ArXiv, abs/1912.11316.

[2] Lee, J., & AlRegib, G. (2020). Gradients as a Measure of Uncertainty in Neural Networks. 2020 IEEE International Conference on Image Processing (ICIP), 2416–2420. doi:10.1109/ICIP40778.2020.9190679

[3] Sun, J., Yang, L., Zhang, J., Liu, F., Halappanavar, M., Fan, D., & Cao, Y. (2022). Gradient-Based Novelty Detection Boosted by Self-Supervised Binary Classification. Proceedings of the AAAI Conference on Artificial Intelligence, 36(8), 8370-8377. https://doi.org/10.1609/aaai.v36i8.20812

[4] Chen, J., Li, J., Qu, X., Wang, J., Wan, J., & Xiao, J. (2023). GAIA: Delving into Gradient-based Attribution Abnormality for Out-of-distribution Detection. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, & S. Levine (Eds.), Advances in Neural Information Processing Systems (Vol. 36, pp. 79946–79958).

[5] Lee, J., Lehman, C., Prabhushankar, M., & AlRegib, G. (2023). Probing the Purview of Neural Networks via Gradient Analysis. IEEE Access, 11, 32716–32732. doi:10.1109/ACCESS.2023.3263210