# Robust Diagrams for Deep Learning Architectures: Applications and Theory

Vincent Abbott Vincent.Abbott@anu.edu.au
Australian National University

July 15, 2024

## 1 Background

Deep learning is a novel field, with truly "deep" algorithms only being possible since the introduction of ResNets in 2015 (1). There has been a proliferation of model architectures to target different modalities of data (2; 3), achieve goals in distinct ways (4; 5), and incorporate new methods to improve model performance and ease of training (6; 7; 8). The details of these configurations are critical to understanding new approaches, implementing models, and innovating on existing designs.

However, there is no standard framework for communicating models. If even one operation is miscommunicated, accurate implementation is impossible. Textual descriptions can easily miss critical details. Prevailing diagrams are ad-hoc and are especially prone to not communicating the axis over which operations act. Attached code can be challenging to interpret and is of varying quality. Often, the actions of a model are hidden behind many layers of abstraction. What a model should do, its high-level structure, and its novel contributions are not communicated with code. Code also depends on the specific style—two algorithms could be almost entirely the same, differing in some vital detail. Yet, if their implementation style is different, this fundamental difference can be laborious to identify. Code, then, is an inefficient means to communicate the details of models.

## 2 The Approach: Formal Diagrams

Formal diagrams overcome these shortfalls. Deep learning algorithms are typically constituted of independent blocks of data, which are organised along axes. Operations act on specific axes. For instance, we can take an operation over rows or columns. These are distinct operations and must be communicated differently. If we can communicate the different blocks of data in memory and the axes over which operations operate, we can communicate all the details of a deep learning model in a clear, comprehensive, and visual manner. This reduces the risk of failing to communicate a key detail, as is often the case with textual or ad-hoc diagrams (See Fig 1). Additionally, we are provided with a conceptual description that can verify code and serve as a framework-independent abstraction for understanding an algorithm. Furthermore, formal algorithms allow for developing and applying robust tools to analyse architectures.

The diagrams used are heavily inspired by category theory. Category theory is the formal mathematics of composition and abstraction and has a rich heritage of diagrammatic schemes

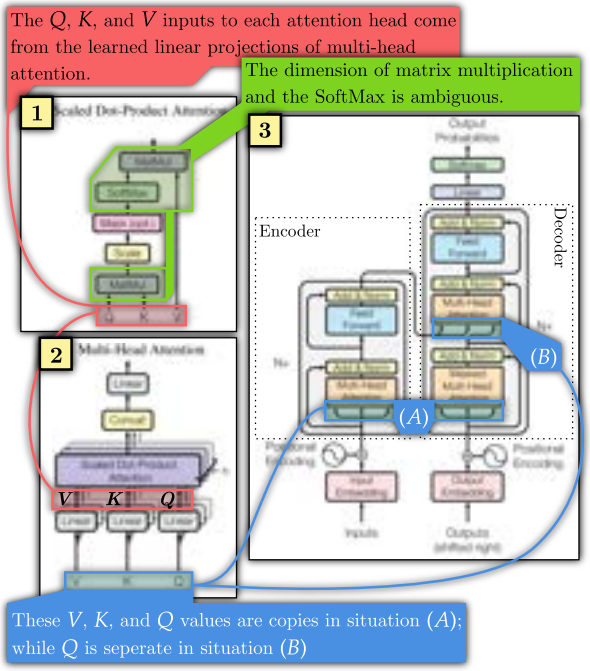


Figure 1: The original transformer diagram from *Attention is All You Need* (2), annotated to highlight unclear details. Critical information is missing regarding the origin of $Q$, $K$, and $V$ values (**red** and **blue**), and the axes over which operations act (**green**).

(9; 10; 11). Neural Circuit Diagrams use many tools from category theory. They can be seen as extensive syntactic sugar applied to the two-category diagrams of Marsden (10; 12) to imitate the rich description of axes provided by (tensor) monoidal string diagrams (9; 13) while also allowing for the manipulation of independent data (Cartesian products).

Dashed lines separate independent data arrays, while solid lines indicate axes. For instance, two lines labelled $x$ and $y$, followed by a dashed line, followed by a line labelled $z$, indicate that we are working with one array of $\mathbb{R}^{x\times y}$ and another of $\mathbb{R}^z$, for total size of $x \times y + z$. This allows us to keep track of the shape of data. Dashed lines separate parallel operations on separate arrays. Operations on specific axes are placed on that axis, making this essential detail of models clear. Taken together, we can clearly show a neural network in Fig 2.

## 3 Completed Work

The initial thesis developed Neural Circuit Diagrams and the tools to display key architectures, including transformers and convolutional image recognition models. The applied aspect has been peer-reviewed and published in *Transactions on Machine Learning Research* (14). Additional architectures have been diagrammed, with significant interest online, showing the interest in robust diagrams. Aspects of their mathematical details were developed in my honours thesis and a preprint (15; 16).

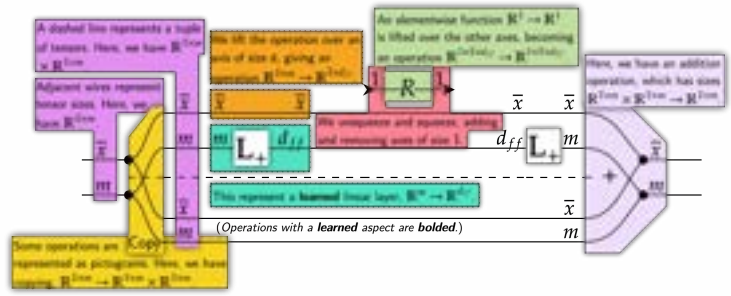


Figure 2: We see that the incoming data of size $\bar{x} \times m$ is copied. A learned linear layer is applied over the $m$ axis in the top section. Overall, this results in the transformation of data of size $\bar{x} \times m$ to $\bar{x} \times d_{ff}$. A line written above an operation represents a SIMD (single instruction, multiple data) operation. The diagram automatically captures the preservation of the $\bar{x}$-sized axis. Element-wise operations such as non-linear activations do not change the data size and hover above the array on which they act. This diagram is sufficient for implementation and captures all the details of the algorithm's behaviour.

# 4 Further Work

The diagrams aim to replace how we communicate deep learning architectures with a formal methodology that can serve as a systematic basis for analysis, innovation, and optimisation. This is the target end state of Neural Circuit Diagrams but will require much further work.

Diagrams aim to be the basis for formal analysis. They are inspired by category theory and can provide a categorical description, offering access to the sophisticated tools of category theory (17; 18). Categorical descriptions also let the probabilistic aspects of deep learning models – such as sampling or quantisation, to be considered (19; 20). However, formalising diagrams to the rigour of categorical diagrams requires further work. Currently, they can be viewed as extensive syntactic sugar applied to categorical diagrams. This syntactic sugar may break the algebraic properties of categorical diagrams (9).

Diagrams contain extensive metadata about algorithms, such as how operations are parallelised into SIMD operations, or whether they can be "fused" – reduced to a low memory algorithm which avoids an intermediate save and load (21; 22). The dominant machine learning framework is PyTorch. It provides access to a library of operations and the automatic generation of backpropagation graphs. However, it does not provide the metadata for automatically fusing and optimising algorithms. This means massive potential gains in performance are missed (21; 22). Formal diagrams can provide this information, offering optimal implementations of machine learning models.

# 5 Acknowledgements

I was supervised by Dr Yoshihiro Maruyama at the Australian National University. This work was supported by JST (JPMJMS2033-02; JPMJFR206P).

# References


[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *CoRR*, vol. abs/1512.03385, 2015. arXiv: 1512.03385.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA* (I. Guyon, U. v. Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, eds.), pp. 5998–6008, 2017.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in neural information processing systems* (F. Pereira, C. Burges, L. Bottou, and K. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," June 2014. arXiv:1406.2661 [cs, stat].

[5] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-Resolution Image Synthesis with Latent Diffusion Models," Apr. 2022. arXiv:2112.10752 [cs].

[6] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *CoRR*, vol. abs/1502.03167, 2015. arXiv: 1502.03167.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part IV* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), vol. 9908 of *Lecture Notes in Computer Science*, pp. 630–645, Springer, 2016.

[8] N. Shazeer, "GLU Variants Improve Transformer," Feb. 2020. arXiv:2002.05202 [cs, stat].

[9] A. Joyal and R. Street, "The geometry of tensor calculus, I," *Advances in Mathematics*, vol. 88, pp. 55–112, July 1991.

[10] D. Marsden, "Category Theory Using String Diagrams," *CoRR*, vol. abs/1401.7220, 2014. arXiv: 1401.7220.

[11] R. Piedeleu and F. Zanasi, "An Introduction to String Diagrams for Computer Scientists," Nov. 2023. arXiv:2305.08768 [cs].

[12] K. Nakahira, "Diagrammatic category theory," July 2023. arXiv:2307.08891 [math].

[13] P. Selinger, "A survey of graphical languages for monoidal categories," Aug. 2009.

[14] V. Abbott, "Neural Circuit Diagrams: Robust Diagrams for the Communication, Implementation, and Analysis of Deep Learning Architectures," *Accepted to Transactions on Machine Learning Research*, July 2023.

[15] V. Abbott, *Robust Diagrams for Deep Learning Architectures: Applications and Theory*. Honours Thesis, The Australian National University, Canberra, Oct. 2023.

[16] V. Abbott and G. Zardini, "Functor String Diagrams: A Novel Approach to Flexible Diagrams for Applied Category Theory," Mar. 2024. arXiv:2404.00249 [math].

[17] B. Fong, D. I. Spivak, and R. Tuyéras, "Backprop as Functor: A compositional perspective on supervised learning," in *34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019*, pp. 1–13, IEEE, 2019.

[18] R. Cockett, G. Cruttwell, J. Gallagher, J.-S. P. Lemay, B. MacAdam, G. Plotkin, and D. Pronk, "Reverse derivative categories," Oct. 2019. arXiv:1910.07065 [cs, math].

[19] T. Fritz, T. Gonda, P. Perrone, and E. F. Rischel, "Representable Markov Categories and Comparison of Statistical Experiments in Categorical Probability," *Theoretical Computer Science*, vol. 961, p. 113896, June 2023. arXiv:2010.07416 [cs, math, stat].

[20] P. Perrone, "Markov Categories and Entropy," *CoRR*, vol. abs/2212.11719, 2022. arXiv: 2212.11719.

[21] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness," June 2022. arXiv:2205.14135 [cs].

[22] T. Dao, "FlashAttention-2: Faster Attention with Better Parallelism and Work Partitioning," Oct. 2023.