Traceoid Memo

Adam Nemecek
adamnemecek@gmail.com

Abstract

This document provides an overview of the company Traceoid, the company behind the computational platform Traceoid. In addition, this document provides a brief overview of the underlying technology. For a thorough overview, please contact us for the Traceoid technical appendix.

Contents

1 Introduction

Traceoid is a computational platform for faster, simpler, interpretable, verified machine learning. Traceoid interprets the entirety of machine learning in the context of convolution, all architectures are interpreted as performing a convolution operation.

Our approach addresses certain problems of current machine learning theories and frameworks, including lack of explainability and speed. Details of our approach are discussed in more detail in our technical

appendix (available upon request).

Our goal is to empower a single developer to be able to develop, train and deploy a machine learning model from scratch.

2 Problems

Despite the staggering advances in machine learning, the field still suffers from several problems.

2.1 Machine learning is expensive

Current approaches to machine learning are capital intensive. Machine learning startups are raising exorbitant amounts of money and spending it on compute. Training ChatGPT-4 cost $100M [Knight (2023)], Anthropic has raised $4B [Weatherbed (2023)] most of which will be spent on compute.

These are just two examples, but [Appenzeller and Bornstein (2023)] indicates that up to 80% of capital raised by ML startups is spent on server costs.

2.2 Model development and deployment is painful

Currently, when developing a model, the developer has to select an architecture from a fixed set of architectures. Different problems require different architectures and each architecture is treated separately, it has specific optimizations, approaches and fine-tunings. These restrictions limit the general applicability of machine learning. What about problems for which it is unclear which architecture to use or for which suitable architectures are nonexistent?

Developing new architectures is a research endeavor that is not realistic when building a product. Experimentation with GPU kernels is also nontrivial and time-consuming.

Deployment is complicated, there are no "deploy and forget" solutions. Scaling training is not automatic. Monitoring of models to prevent model degradation is not automatic. Managing training pipelines is complicated.

2.3 Models are black boxes

Currently, machine learning models lack:

- **Interpretability**: Can we understand the behavior of the models? Interpretability is the Holy Grail of machine learning.
- **Debuggability**: Can we fix models when they go wrong?
- **Verifiability**: Can we guarantee behavior of the models?

Due to these problems, machine learning cannot be deployed in dependable contexts.

3 Traceoid

Traceoid is a machine learning platform [1] which is based on several principles.

- **Unified**: All machine learning architectures, including convolutional neural networks, diffusion models, and transformer models are cast as convolutions in the context of a Hopf algebra. This leads to a radical simplification of every aspect of the machine learning process.

[1] integrated framework and hosting

- **Geometric**: Machine learning is inherently geometric. Interpreting these geometries solves the black box problem. Furthermore, more structure is extracted during the training process which leads to faster training.
- **Integrated**: Traceoid provides all functionality the developer might need out of the box. Integration between framework and hosting platform simplifies deployment.

3.1 Unification via convolution

Traceoid is built around the idea that the entirety of machine learning can be cast in the context of convolution and convolution integral equations.

Given the convolution transform

$$conv(input, weights) = output$$

.

inference amounts to calculating the output given input and weights, while training amounts to finding weights from inputs and outputs.

There is emerging evidence that the convolutional interpretation is correct.

- **Convolutional neural networks**
  - Obvious connection.
- **Transformers**:
  - [Andreoli (2020)] discusses the attention mechanism of transformer models as deformable convolution.
  - [Poli et al. (2023)] replaces the attention mechanism with the standard convolution operation to achieve performance gains. Our approach is similar only superficially, our approach generalizes to all architectures, not just transformers, and also the convolution employed by this line of research is not the correct one. A more suitable convolution can yield a significant speedup. In addition, our approach does not incur any trade-offs.
- **Diffusion models**:
  - The connection between convolution, differentiation and diffusion has been understood for a long time [Sousa et al. (2022)].

Treating all architectures as convolutions allows us to speed up both the training and inference process using transformations similar to the Fourier transform. The Fourier transform is a transform used to speed up convolution and is possibly the most important numerical algorithm. Fourier transform and its variants are ubiquitous and are used in the context of signal processing (including audio, image and video processing), time series, electrical engineering etc. For example, the Radon transform, a related transform, is used in the context of computer tomography (CT scan) to reconstruct a 3D image of the scanned object from single slices [Epstein (2008)].

We employ an esoteric variant of the Fourier transform which we use to speedup an esoteric variant of the convolution operation.

The advantage of our approach is the universality. Our approach speeds up all architectures, not just a particular architecture. For transformers, our approach achieves subquadratic attention, without any trade-offs.

Convolutional interpretation also answers open questions about superpositions observed in the context of neural networks posed for example in [Elhage et al. (2022)]. Superposition and convolution are closely related concepts, convolution is a superposition of shifted signals.

3.2 Machine learning as a geometry discovery process

Training a model amounts to finding decision boundaries inherent in the data. These decision boundaries can be interpreted geometrically as hyperplanes.

Combining these hyperplanes allows us to interpret weights as geometric objects called polytopes.

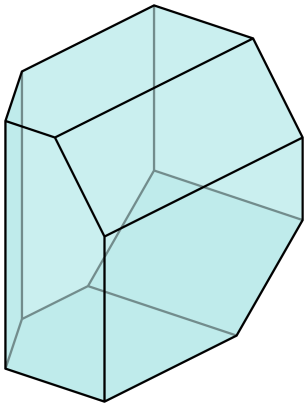

Figure 1: Polytope [Wikipedia: Simple polytope (n.d.)]

The connection between hyperplanes, and polytopes in the context of machine learning is discussed for example in [Black et al. (2022)], [Cattell (2016)], [Masden (2022)], [Huchette et al. (2023)] and [Hanin and Rolnick (2019)]. In addition, formation of polytopes in the context of model weights has been observed in the wild by Anthropic as discussed in [Elhage et al. (2022)].

Our approach makes these polytopes explicit and enables explainability, and verification. Interpreting these polytopes as zonotopes allows us to use techniques which have been used in the context of verified numerical computation [Girard (2005)]. Fundamentally, zonotopes allow us to divide the input space into single clusters which represent states. Transitions between single states can then be used to answer questions about the validity and safety of the model's behavior.

Note that use of zonotopes in the context of verified machine learning has been previously explored for example by [Mirman et al. (2018)] and [Gehr et al. (2018)] with success. Traceoid offers a complete, integrated product as well as faster verifications.

4 Hopf algebra

Hopf algebra is a rich algebraic structure, a generalization of tensor algebra which is equipped with the notion of an antipode. The antipode corresponds to the convolutional inverse [Heunen (2013)], and as such, it corresponds to the weights of the model.

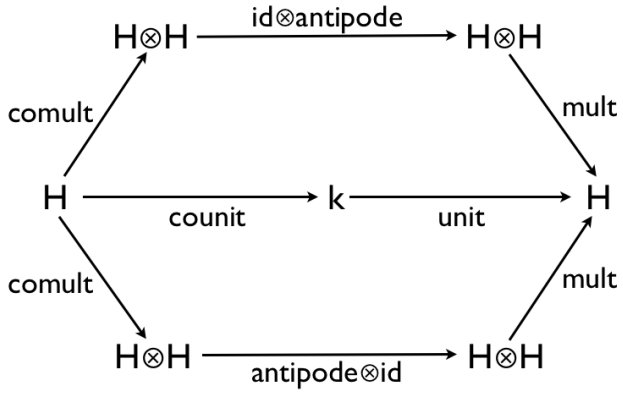

Figure 2: Hopf algebra [Amos (2023)]

The top path [2] and bottom path [3] of the algebra are convolutions, generalizations of the standard convolution operation. The Hopf algebra has the property that it calculates the antipode, the weights, while calculating the convolution operation. This process corresponds to *learning*.

There is a path from Hopf algebras to neural networks. Hopf algebras are used in the context of renormalization in quantum field theory [Broadhurst and Kreimer (1998)] which itself can be interpreted as a foundation of machine learning [Roberts et al. (2021)].

Renormalization can be modeled using the Ising model [Battle (1999)] which itself has an interpretation as a recurrent neural network via Hopfield networks [Rojas (1996)].

Hopfield networks have been recently demonstrated to be equivalent to transformer models [Ramsauer et al. (2020)].

[Ambrogioni (2023)] establishes an equivalence between Hopfield networks and diffusion models.

5 Business

Traceoid will be an open-core product like Apache Spark developed by Databricks, i.e. the main product will be open source while additional enterprise functionality will be paid.

Furthermore, we are exploring the possibility of a Shopify-like plugin ecosystem. Developers will make money by building models that are then used by other parties on the platform. Traceoid will get a cut of the transactions.

[2] $comult- > id \otimes antipode- > mult$
[3] $comult- > antipode \otimes id- > mult$

6 Market

The machine learning market in 2023 is estimated at $50B, and is projected to grow to $771B over the next 10 years.

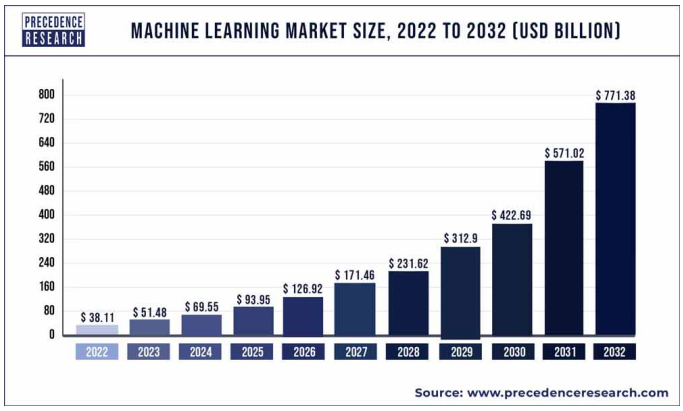

Figure 3: ML Market [Precedence Research (2023)]

We believe that the dominant solution, possibly Traceoid, can capture 40% of the market.

7 Competitors

Since Traceoid is a framework, a hosting platform, and a general approach to machine learning, our competitors include frameworks, hosting platforms, tools, as well as others.

- **Frameworks**: Pytorch, Modular Mojo
- **Cloud hosts**: AWS, Google Cloud
- **Hybrids**: Julia & JuliaHub
- **Tools**: HuggingFace
- **Explainability**: xAI
- **Better transformer perf**: Cartesia.ai

While one can train and deploy a model using a combination of these, the entire process is still a dark art, especially if one is developing and training a model from scratch. The process is time-consuming, there is very little in terms of guidance. Overall, it is not a single person operation

We posit that developers follow the best tools. The launch of the Apple app store brought a deluge of new iOS developers, and a deluge of apps as a result. Eventually, Apple's marketing slogan "there's an app for that" became true as a result.

The machine learning ecosystem is still waiting for this moment. ML applications are still somewhat rare, the "there's a model for that" moment is yet to come.

We are replicating Apple's strategy. By building an integrated solution which makes it easy for developers to build a model and use it to generate income, we hope to attract developers in droves.

The goal of Traceoid is the following: **Allow a developer with zero machine learning experience to learn machine learning, develop, train and deploy a model in 6 months**.

7.1 Differentiators

In addition to the advantages of general speedup, explainability, cheaper & faster training, and ease of deployment, Traceoid also removes explicit architectures. Due to a duality between weights and architectures, both the architecture and weights are discovered during the training process.

8 Venture round

Traceoid is raising $1.5M pre-seed. The majority of this money will be spent on salaries. In the first year (possibly faster), we will develop a prototype of our approach in the Rust programming language. This prototype will run on the CPU and we will benchmark it against other CPU-based implementations. Following this, we will implement our approach on the GPU.

Throughout this process, we will focus on establishing developer relations and building a community.

9 Biographies

- **Adam Nemecek**: is the founder of Traceoid. Adam has single-handedly implemented macOS recording for loom.io which has been recently acquired by Atlassian for ~$1B. Previously, he has worked at Microsoft. Adam graduated from Harvard with a bachelor's degree in Computer Science.
- **Ammar Husain**: obtained a PhD in Topological Quantum Computation from Berkeley and has up until recently worked as a quantum compiler developer at Horizon Quantum.
- **Iago Leal de Freitas**: Iago has graduated with an MSc from Federal University of Rio de Janeiro, one of Brazil's top universities. Until recently he worked as a programmer/mathematician for a Brazilian energetics consulting company.

References

Ambrogioni, L. (2023). In search of dispersed memories: Generative diffusion models are associative memory networks.

Amos, D. (2023, March 3). What is a Hopf algebra?

Andreoli, J.-M. (2020). Convolution, attention and structure embedding.

Appenzeller, G., and Bornstein, M. (2023, April 27). Navigating the High Cost of AI Compute.

Battle, G. (1999). Wavelets and renormalization. *Wavelets and Renormalization*.

Black, S., Sharkey, L., Grinsztajn, L., Winsor, E., Braun, D., Merizian, J., ... Leahy, C. (2022). Interpreting neural networks through the polytope lens.

Broadhurst, D. J., and Kreimer, D. (1998). Renormalization automated by hopf algebra.

Cattell, S. (2016). Geometric decomposition of feed forward neural networks.

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., ... Olah, C. (2022). Toy models of superposition.

Epstein, C. L. (2008). *Introduction to the mathematics of medical imaging*. *Introduction to the Mathematics of Medical Imaging*.

Gehr, T., Mirman, M., Drachsler-Cohen, D., Tsankov, P., Chaudhuri, S., and Vechev, M. (2018). AI2: Safety and robustness certification of neural networks with abstract interpretation. *Proceedings - IEEE Symposium on Security and Privacy*, *2018-May*, 3–18.

Girard, A. (2005). Reachability of uncertain linear systems using zonotopes. In *Lecture Notes in Computer Science*,Vol. 3414.

Hanin, B., and Rolnick, D. (2019). Deep ReLU networks have surprisingly few activation patterns.

Heunen, C. (2013). *Quantum physics and linguistics*. (C. Heunen, M. Sadrzadeh, & E. Grefenstette, Eds.). Oxford University Press.

Huchette, J., Muñoz, G., Serra, T., and Tsay, C. (2023). When deep learning meets polyhedral theory: A survey.

Knight, W. (2023, April 17). OpenAI's CEO Says the Age of Giant AI Models Is Already Over.

Masden, M. A. (2022). Algorithmic determination of the combinatorial structure of the linear regions of ReLU neural networks.

Mirman, M., Gehr, T., and Vechev, M. (2018). Differentiable abstract interpretation for provably robust neural networks.

Poli, M., Massaroli, S., Nguyen, E., Fu, D. Y., Dao, T., Baccus, S., ... Ré, C. (2023). Hyena hierarchy: Towards larger convolutional language models.

Precedence Research. (2023). *Machine Learning Market*. Retrieved from http://precedenceresearch.com/machine-learning-market

Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., ... Hochreiter, S. (2020). Hopfield networks is all you need.

Roberts, D. A., Yaida, S., and Hanin, B. (2021). The principles of deep learning theory.

Rojas, R. (1996). *Neural networks: A systematic introduction*. *Neural Networks*,Vol. 7.

Sousa, R., Guerra, M., and Yakubovich, S. (2022). Convolution-like structures, differential operators and diffusion processes. In *Lecture Notes in Mathematics*,Vol. 2315.

Weatherbed, J. (2023). Amazon will invest up to $4 billion into OpenAI rival Anthropic.

Wikipedia: Simple polytope. (n.d.). Simple polytope.