Journal of Science and Technology
ISSN: 2456-5660 Volume 10, Issue 12 (Dec -2025)
www.jst.org.in
DOI: https://doi.org/10.46243/jst.2025.v10.i12.pp01-10

The Logic and the Ledger: Tracing the DNA of AI

Rajendran Swamidurai[1] | Uma Kannan[2]

[1]Department of Math and Computer Science, Alabama State University
[2]Department of Math and Computer Science, Alabama State University

To Cite this Article
Rajendran Swamidurai and Uma Kannan, "The Logic and the Ledger: Tracing the DNA of AI", *Journal of Science and Technology*, Vol. 10, Issue 12, December 2025, pp.01-10.

Article Info
Received: 02-Aug-2025 Revised: 22-Aug-2025 Accepted: 03-Sep-2025 Published: 15-Dec-2025

ABSTRACT

The development of intelligent systems is fundamentally and inseparably linked to a sophisticated mathematical framework. Modern artificial intelligence (AI), particularly its subfields of machine learning (ML) and deep learning, is not a new discipline of computer science but rather a highly advanced application of classical and novel mathematical principles. The models and algorithms that enable systems to process data, learn intricate patterns, and optimize predictions are built upon a bedrock of abstract mathematical theories. This paper systematically deconstructs this relationship, demonstrating how core mathematical disciplines serve as the language, the engine, and the conceptual framework for all intelligent systems. The analysis will traverse from foundational principles to their application in cutting-edge architectures and conclude with a discussion of the theoretical and practical challenges that are currently shaping the future of the field.

KEYWORDS: *Artificial Intelligence, Mathematical Backbone of AI, Deep Learning, Machine Learning, Large Language Models.*

I. INTRODUCTION

The field of artificial intelligence was formally christened at the 1956 Dartmouth Summer Research Project on Artificial Intelligence [2]. From its inception, a core philosophical divide emerged between two competing paradigms. The first, known as the Symbolic or "Good Old-Fashioned AI" (GOFAI) school, was rooted in formal logic and rule-based systems [4]. This approach was aligned with the rationalist view of the mind, positing that intelligence could be achieved by representing knowledge and reasoning through symbolic abstractions and explicit rules [4]. Early triumphs included expert systems, such as SAINT, a program developed in 1961 that could solve elementary symbolic integration problems at the level of a college freshman [3]. This paradigm, which relied on heuristic search to find solutions in large combinatorial spaces, was prevalent from the 1950s to the 1970s and was marked by an overemphasis on Boolean (True/False) logic [2].

The second paradigm, the Connectionist school, was inspired by the biological neural networks of the human brain [4]. Rooted in an empiricist philosophy, this approach focused on building systems that could learn from data and infer patterns without explicit instructions or prior knowledge [4]. Early work by Frank Rosenblatt on the Perceptron in the 1950s laid the groundwork, and the field was formally established in the 1980s by researchers such as David Rumelhart [2]. Despite this promising start, the field of AI experienced a significant downturn in the 1970s, now known as the AI winter, due to a gap between theoretical promise and practical success [2].

The resurgence of AI in the 1990s and 2000s, driven largely by the connectionist and statistical approach, was not merely a philosophical shift but a direct consequence of a fundamental change in computational and economic reality. The early symbolic approach was computationally manageable in an era of limited resources [2].

Journal of Science and Technology
ISSN: 2456-5660 Volume 10, Issue 12 (Dec -2025)
www.jst.org.in
DOI: https://doi.org/10.46243/jst.2025.v10.i12.pp01-10

However, the later decades of the twentieth century saw the emergence of the internet, which enabled the gathering of large amounts of data, and the widespread availability of computational power and storage to process that data [2]. This technological enablement provided the essential fuel for data-driven, statistical methods to flourish, allowing them to overcome the limitations of explicit rule-based systems. This causal relationship—where the economic and infrastructural feasibility of data processing dictated the dominant theoretical and philosophical paradigm—demonstrates that the modern dominance of statistical AI is not a testament to a purely ideological victory but a practical one, driven by the affordances of the digital age.

Linear Algebra: The Language of Data and Representation

At its core, linear algebra serves as the language and computational backbone of modern AI [7]. The field provides the mathematical framework for handling and manipulating multidimensional data efficiently [8]. This begins with the fundamental building blocks of data representation: scalars (single numbers), vectors (ordered arrays with magnitude and direction), matrices (two-dimensional arrays), and tensors (multi-dimensional arrays) [7]. For instance, a dataset of house prices can be represented as a matrix where each row is a house and each column is a feature such as size or number of bedrooms [8]. Similarly, an image can be transformed into a matrix of pixel values [7].

Beyond mere representation, linear algebra provides the tools for transforming and analyzing this data. Matrix operations are essential for processing vast datasets efficiently, while linear transformations are used in deep learning for tasks like scaling and rotating data [8]. Two of the most critical concepts are eigenvalues and eigenvectors, which are foundational for dimensionality reduction techniques like Principal Component Analysis (PCA) [1]. PCA uses these concepts to identify the most significant features in a dataset, thereby reducing its complexity without compromising performance [1]. Furthermore, the use of tensors, which generalize vectors and matrices, is what enables neural networks to process complex, multi-dimensional data like images and text [7].

Calculus: The Engine of Learning and Optimization

While linear algebra provides the structure for data, calculus provides the operational engine that allows intelligent systems to learn and improve [7]. The core principle here is the concept of derivatives and gradients, which measure how a function changes with respect to its input variables [8]. In the context of AI, this function is typically a "loss function" that quantifies the error between a model's predictions and the actual values [11].

The central algorithm that leverages this principle is Gradient Descent [1]. This iterative optimization algorithm works by minimizing the loss function by continuously adjusting a model's parameters (weights and biases) [11]. It does this by taking a step in the direction opposite to the gradient, which corresponds to the steepest descent on the loss function's surface [11]. Variants of this fundamental algorithm, such as Stochastic Gradient Descent (SGD) and Mini-Batch Gradient Descent, improve efficiency by updating parameters on smaller subsets of data, making them more suitable for large datasets [12]. More advanced optimizers like RMSProp and Adam, which are built upon these principles, further refine the training process to improve convergence speed and performance [7].

Complementing gradient descent is Backpropagation, a mechanism that is critical to the training of neural networks [11]. While gradient descent is the algorithm for descending the cost function, backpropagation is the method for calculating the gradients needed for this descent [13]. It works by propagating the error backward from the output layer to the input layer, using the chain rule of calculus to efficiently compute the partial derivative of the cost function with respect to each weight and bias in the network [7]. The two algorithms work in concert, with backpropagation providing the gradient information that gradient descent uses to update the model's parameters [13].

Probability and Statistics: The Framework for Uncertainty and Prediction

In a world of imperfect and noisy data, probability and statistics provide the essential framework for AI to handle uncertainty, make informed predictions, and validate its models. Statistical techniques such as Bayesian analysis, hidden Markov Models (HMMs), and Gaussian Mixture Models (GMMs) are used in applications

Journal of Science and Technology
ISSN: 2456-5660 Volume 10, Issue 12 (Dec -2025)
www.jst.org.in
DOI: https://doi.org/10.46243/jst.2025.v10.i12.pp01-10

like speech recognition and natural language processing to model variability and uncertainty. Probability theory gives AI systems the tools to make predictions with incomplete information, with Bayesian networks, for example, modeling probabilistic relationships between variables to handle noisy data. [1]

These disciplines also provide the tools for robust model validation. Techniques such as correlation analysis, confidence intervals, cross-validation, and hypothesis testing are used to isolate the most relevant features, assess model accuracy, and quantify the uncertainty of predictions. Parameter estimation methods like Maximum Likelihood Estimation (MLE) and Maximum a Priori estimation (MAP) are also used in training algorithms to improve model accuracy. [1]

The four foundational fields—linear algebra, calculus, probability, and statistics—are not independent modules but a deeply integrated and synergistic toolkit. A model is rarely a pure application of a single discipline; rather, it is a complex synthesis of multiple fields. For example, a simple linear regression analysis combines concepts from both linear algebra and probability, while the process of training such a model uses calculus-based optimization (gradient descent) to minimize a loss function, which is itself a statistical concept. [1] This interconnectedness means that a thorough understanding of one field is often contingent on a working knowledge of the others, making their combined mastery a prerequisite for building truly intelligent systems.

Table 1: Foundational Math Disciplines and AI Applications

Discipline	Data Representation	Optimization & Learning	Uncertainty Management	Model Validation & Analysis
Linear Algebra	Vectors, Matrices, and Tensors represent data points and features; data sets are often represented as matrices [7]	Matrix multiplication and transformations enable neural network computations and data processing [8]	Covariance matrices model relationships between variables in probabilistic settings [1]	Principal Component Analysis (PCA) uses eigenvalues and eigenvectors for dimensionality reduction [1]
Calculus	N/A	Gradient Descent minimizes loss functions by adjusting parameters using partial derivatives [1]	Integrals are used to compute probabilities, expected values, and cumulative distributions in probabilistic models [8]	Hessian matrices and curvature are used to understand the local geometry of the loss function landscape [14]
Probability	Probability distributions (e.g., Gaussian, Binomial) are used to represent and predict outcomes [8]	Bayesian Inference updates probabilities based on new data to enable adaptive learning [8]	Bayesian networks model uncertainty and probabilistic relationships between variables [1]	Hypothesis testing, cross-validation, and confidence intervals are used to quantify prediction uncertainty [1]
Statistics	Datasets are framed as populations or samples to be analyzed [1]	Maximum Likelihood Estimation (MLE) and Maximum a Priori (MAP) are used to train algorithms and improve accuracy [1]	Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs) are used in applications with variability and uncertainty [1]	Correlation analysis and exploratory data analysis (EDA) isolate relevant features and quantify patterns [1]

II. Mathematical Principles in AI Architectures

Neural Networks: The General Mathematical Framework

Artificial Neural Networks (ANNs), the foundational architecture of deep learning, are computational systems loosely inspired by the biological brain [6]. They are built upon interconnected layers of "neurons" or nodes, each of which performs a specific mathematical operation [15]. These operations are defined by parameters called weights and biases, and their output is passed through an activation function before being transmitted to the next layer [15]. A crucial aspect of this design is that these activation functions are nonlinear [15]. This nonlinearity is what enables neural networks to model the complex, non-linear patterns and dependencies found in real-world data, far beyond the capabilities of simpler linear models [15]. The training process of a neural network is an iterative, mathematical cycle of computing the error between its predictions and the actual values and then adjusting the internal weights and biases to reduce that error [11].

Convolutional Neural Networks (CNNs): The Power of Convolution for Vision

Convolutional Neural Networks (CNNs) are a specialized type of neural network that has achieved remarkable success in computer vision and other fields that involve grid-like data [1]. The distinguishing feature of a CNN is the mathematical operation of convolution, which is applied in its convolutional layers [15]. In a convolution, a small matrix called a "filter" or "kernel" is applied to an image, which is itself a larger matrix of pixel values [15]. The filter moves across the image, performing matrix multiplication and addition at each position to extract important features such as edges, lines, and textures [8].

The mathematical design of a CNN exploits a fundamental property of visual data: translation equivariance. In a traditional neural network, every neuron is connected to every pixel, which would require an immense number of parameters for a high-resolution image. CNNs, by contrast, use a small, local filter that "looks at small areas of the image one by one," drastically reducing the number of parameters and computational overhead. [16] The repeated application of the same filter across the entire image means that the model can detect a feature regardless of its position [18]. This architectural choice, which embeds the geometric property of translation into the model's logic, is a prime example of how a specific mathematical design can lead to vastly improved efficiency and performance by aligning the algorithm with the underlying structure of the data it is designed to process.

Recurrent Neural Networks (RNNs): Modeling Sequence and Memory

Recurrent Neural Networks (RNNs) are a class of neural networks specifically designed to handle sequential data, such as text and time series [1]. Unlike conventional feedforward networks that map a single input to a single output, RNNs process a sequence of inputs through a recurrent loop [15]. The core mathematical concept that enables this is the hidden state, which acts as a form of internal "memory" that captures information from previous steps in the sequence [15]. At each time step t, the hidden state st is calculated based on the current input xt and the hidden state from the previous time step, st−1. This relationship is expressed mathematically by the update equation: st=f(Uxt+Wst−1), where f is a nonlinear activation function and U and W are weight matrices that do not change for an entire sequence. [20] The hidden state carries information through the sequence, allowing the network to understand context and order, which is critical for tasks like natural language processing. [15] This reliance on matrix operations to process sequential data highlights the central role of linear algebra in building models that understand context and temporal relationships.

The Attention Mechanism: The Mathematical Leap Behind LLMs

While RNNs introduced the concept of memory for sequential data, they suffered from a key weakness: they favored more recent information and tended to "attenuate" or lose context from earlier parts of a long sequence. The attention mechanism was developed to solve this problem by enabling a token to have "equal access to any part of a sentence directly, rather than only through the previous state". [21]

Mathematically speaking, an attention mechanism computes "attention weights" that reflect the relative importance of each part of an input sequence to the task at hand [22]. This process involves a series of critical mathematical operations. First, for each token in the input sequence, three vectors are created: a query vector, a key vector, and a value vector [22]. The core of the operation lies in the dot product between the query vector of a given token and the key vector of every other token in the sequence [21]. This operation quantifies the "alignment" or relevance between the token seeking information and the information contained in all other tokens [22]. These dot product scores are then scaled and passed through a softmax function, which normalizes them into a set of attention weights that sum to one [21]. This results in a probability distribution, where each token's vector is updated based on a weighted average of the value vectors of all other tokens, with the weights being the attention scores [22].

This architectural shift, enabled by a change in mathematical design, is the primary reason for the emergence of Large Language Models (LLMs) and their unprecedented scale. Unlike RNNs, which process sequences in a slow, step-by-step fashion, the attention mechanism relies on highly parallelizable matrix operations [21]. This allows the model to calculate the relationships between all tokens in a sequence simultaneously, a crucial distinction that allows for training on massive, terabyte-scale datasets that would be computationally infeasible for a sequential architecture [23]. The result is a direct causal link between a specific mathematical innovation and the emergence of an entirely new class of AI systems.

III. Addressing Theoretical Challenges and Limitations

The Generalization Gap: Understanding Model Performance Beyond Training

A core theoretical challenge in AI is the generalization gap, defined as the discrepancy between a model's performance on its training data and its performance on new, unseen data from the same distribution. While traditional statistical theory suggests that model performance should decrease as complexity increases beyond a certain point, empirical evidence in deep learning points to a counterintuitive phenomenon known as the double descent curve. This behavior indicates that in "over-parameterized" models—where the number of parameters far exceeds the number of training examples—the generalization gap paradoxically decreases as model complexity grows. [27]

A deeper understanding of this phenomenon requires analyzing the loss landscape, a high-dimensional surface where each point corresponds to a model's parameters and its associated loss value [30]. Research in this area contrasts sharp minima with flat minima in this landscape. Models that converge to flat minima tend to generalize better because they are more robust to small perturbations in the data or parameters [30]. The reason for this lies in the geometry of the loss function, where flat minima correspond to a larger basin of attraction [30]. For example, studies have shown that Stochastic Gradient Descent (SGD) tends to find these flat minima, which leads to better generalization [30]. Researchers are now developing new mathematical tools to characterize this problem, such as Functional Variance, which is a concept from Bayesian learning that provides an asymptotically unbiased estimator for the generalization gap in over-parameterized settings where traditional methods fail [27].

Model Interpretability and Explainable AI (XAI)

As AI models become more complex and are deployed in high-stakes domains like finance and healthcare, their "black box" nature has become a significant problem [31]. The lack of transparency makes it difficult for humans to understand how and why a model makes a specific prediction [32]. This challenge is driven by two main factors: ethical and practical concerns, such as the need to debug models, identify biases, and ensure they adhere to industry best practices, and regulatory pressures that require systems to provide "understandable explanations" to data subjects [32].

To address this, the field of Explainable AI (XAI) is developing new mathematical frameworks to provide clarity into these complex systems. Key methods include: 1) LIME (Local Interpretable Model-agnostic Explanations): This technique works by creating a simpler, more interpretable model (such as a linear model or decision tree) to approximate the behavior of a complex, black box model around a single prediction [14] . It provides a local, understandable explanation for a single result, 2) SHAP (Shapley Additive Explanations): This method is based on Shapley values from cooperative game theory [14]. It fairly attributes the contribution of each input feature to a model's output by considering all possible combinations of features [14]. This provides both local and global interpretability.

A more foundational approach to model interpretability is Dictionary Learning, which seeks to deconstruct a model's internal representations from first principles. The central equation of this research is $Y \approx DX$. In this formulation, Y represents a model's messy, superimposed internal representations, D is a "dictionary" of pure, monosemantic features, and X is a "sparse code" or recipe that explains how to reconstruct Y from a handful of features in D. [36]

The complexity of modern AI models creates a causal chain that connects technical challenges with real-world pressures. The mathematical complexity that enables high performance also creates a generalization gap, and the black box nature of these models makes it nearly impossible to diagnose the sources of this poor generalization or to identify systemic biases. [32] This is further compounded by societal and legal demands for accountability and transparency. The need for greater interpretability is therefore not just a technical luxury but a critical requirement, driven by a complex feedback loop of theoretical limitations, real-world failures, and regulatory imperatives.

IV. The Future of Mathematics in AI

Beyond Euclidean Spaces: The Rise of Geometric Deep Learning

Traditional deep learning models such as CNNs and RNNs are designed for data that exists on regular grids or sequences, known as Euclidean data. However, a vast amount of real-world data, from social networks to molecular structures, exists in complex, non-Euclidean spaces like graphs, manifolds, and point clouds. This has given rise to Geometric Deep Learning (GDL), a burgeoning field that extends the capabilities of neural networks to handle these irregular data types by incorporating geometric and topological principles. [19]

GDL addresses the "curse of dimensionality" by leveraging known symmetries and invariances in data, such as rotation or translation [18]. By encoding these physical properties directly into the model's architecture, GDL systems can learn more efficiently and accurately [19]. A core mathematical principle in this field is topology, the study of shapes and spatial properties [38]. Topological methods, such as persistent homology, allow GDL models to capture "higher-order relationships" and the overall "shape" of data, making them more robust to noise and perturbations [37]. This approach represents a fundamental synthesis of the symbolic and connectionist paradigms: it is a return to a "first principles" approach, where prior knowledge about the structure of the data is used to design and constrain a data-driven model, resulting in a more principled and efficient system.

Category Theory: A New Language for Abstraction and Composition

Category theory, a branch of mathematics that provides a high-level framework for understanding structure and relationships, is an emerging frontier in AI research. It offers a potential bridge between the symbolic and sub-symbolic approaches to AI, providing a unified language for describing and comparing different paradigms, from deep learning to reinforcement learning. By framing neural network components as categorical objects and morphisms, it may lead to the development of new, more modular, flexible, and interpretable architectures. The ability to reason about complex systems in an abstract and general way is what makes category theory a powerful tool for exploring the fundamental building blocks of intelligence. [40]

The Symbiotic Loop: AI as a Tool for Mathematical Discovery

The relationship between mathematics and AI is not a one-way street; AI is now being used to accelerate and advance mathematics itself [10]. This has led to breakthroughs in several areas: 1) Automated Theorem Proving: AI algorithms are now capable of proving complex mathematical theorems by efficiently exploring vast mathematical spaces [10]. Generative AI can assist in this process by translating human-written proofs into a format that computers can verify, a method that is described as "solving one problem with another" [42], 2) Discovery of New Concepts: AI models, trained on large datasets of mathematical structures, can recognize patterns and regularities that have "eluded human mathematicians" and generate novel expressions and structures [10], and 3) Solving Open Problems: AI has already contributed to solving decades-old problems. For example, Google DeepMind's AlphaProof has performed at the level of a silver medalist in the International Mathematical Olympiad, while a Caltech team used an AI algorithm to disprove potential counterexamples to the 60-year-old Andrews–Curtis conjecture, a problem in group theory [42].

This application of AI to mathematics represents a profound causal feedback loop. A tool built on the language of mathematics is now being used to advance that very language, extending the boundaries of abstract and creative thought. The ability of AI to navigate vast, complex problem spaces and find unexpected solutions suggests that it is on the cusp of becoming a genuine partner in the most fundamental intellectual pursuits.

The Intersection of Theory and Practice: Computational Demands and Architectural Trends

The mathematical complexity of AI models has a direct and profound impact on their computational demands and, by extension, on industry trends and business strategy. A clear distinction can be drawn between traditional machine learning (Classic ML) and large language models (LLMs): 1) Classic ML algorithms, such as linear regression or decision trees, are typically simpler in design. They often contain between 1 and 10 million parameters, are well-suited for structured and smaller datasets, and can be run on low-end machines with less computational power [24]. They are ideal for specific, well-defined problems and offer a high degree of transparency and interpretability [24], and 2) LLMs, by contrast, are built on advanced architectures like Transformers. They are vastly more complex, with billions or even trillions of parameters [23]. They are data-hungry and require significant computational resources, often necessitating specialized hardware like GPUs or TPUs and distributed computing environments [23]. LLMs are best suited for complex, general-purpose tasks involving unstructured data [24].

The immense computational cost associated with training and maintaining LLMs is not merely a technical bottleneck but a strategic liability. This has led to the emergence of new, resource-conscious architectures like Mixture-of-Experts (MoE) models, which intelligently allocate resources to be more cost-efficient at scale [31]. The high cost and "black box" nature of massive models are also pushing businesses toward "stack ownership," where they design architectures that are more auditable, traceable, and easier to control in-house [31]. The mathematical complexity of these systems is therefore directly driving a market shift toward smaller, smarter, and more compliant systems.

Table 2: The Mathematical and Computational Divide

Feature	**Classic Machine Learning**	**Deep Learning**	**Large Language Models (LLMs)**
Model Complexity	1 to 107 parameters [17]	105 to 108 parameters (varies widely)	109 to 1012 parameters or more [23]
Key Mathematical Operations	Linear regression, matrix operations, statistical modeling [1]	Gradient descent, backpropagation, nonlinear activation functions [8]	Attention mechanism, parallelizable matrix operations [21]
Data Requirements	Typically requires structured, labeled data [24]	Requires large datasets for training, often structured [41]	Relies on massive, unstructured datasets (e.g., text, video) [24]
Computational Demands	Lower. Can run on low-end machines [17]	Substantial. Training requires specialized hardware like GPUs or TPUs [25]	Very high. Requires multiple parallel processing units and extensive GPU resources [23]
Interpretability	Generally easier to interpret and analyze [24]	Can be difficult to explain due to "black box" nature [26]	"Black box" models with a significant lack of transparency [26]

V. Conclusion

The evidence overwhelmingly suggests that mathematics is not merely a tool for AI but its fundamental language, its engine of operation, and its core conceptual framework. The historical journey from symbolic logic to statistical learning was propelled not just by philosophical debate but by the practical feasibility afforded by advances in computational infrastructure. Modern AI models are a complex synthesis of multiple mathematical disciplines, with linear algebra providing the data's structure, calculus enabling the learning and optimization process, and probability and statistics serving as the framework for prediction and validation.

The intricate architectures of deep learning, from CNNs to LLMs, are a testament to the power of applied mathematical principles. The translation equivariance embedded in CNNs and the parallelizable nature of the attention mechanism in LLMs are prime examples of how a specific mathematical design can lead to breakthroughs in efficiency and performance. However, this increased complexity has introduced significant theoretical challenges, such as the generalization gap and the lack of model interpretability. The response to these challenges is also deeply mathematical, with fields like XAI and GDL developing new frameworks to build more robust, transparent, and principled systems.

Ultimately, the relationship is a symbiotic loop. While mathematics provides the foundation for AI, AI is now becoming a partner in advancing mathematics itself, solving open problems and discovering new concepts that have eluded human researchers for decades. The most transformative breakthroughs will not be isolated technical feats but will emerge from the dynamic and deepening synergy between mathematical theory and AI application, as the two fields continue to co-evolve in a collaborative and creative cycle.

Acknowledgments

This research was funded by the National Science Foundation (NSF), grant numbers 2406117.

References

[1] Jeff Shepard (2025). What is the mathematics behind artificial intelligence?. *EEWorld Online*, https://www.eeworldonline.com/what-is-the-mathematics-behind-artificial-intelligence/

[2] Standford University (2016). One Hundred Year Study on Artificial Intelligence (AI100): Appendix I: A Short History of AI. https://ai100.stanford.edu/2016-report/appendix-i-short-history-ai

[3] Lawrence Livermore National Security. The birth of Artificial Intelligence (AI) research. Science and Technology. https://st.llnl.gov/news/look-back/birth-artificial-intelligence-ai-research

[4] Ashok K. Goel (2021). Looking back, looking ahead: Symbolic versus connectionist AI. *AI Magazine*. 2021;42:83–85, DOI: 10.1609/aaai.12026

[5] Jieshu's Blog. Symbolism vs. Connectionism: A Closing Gap in Artificial Intelligence. http://wangjieshu.com/2017/12/23/symbol-vs-connectionism-a-closing-gap-in-artificial-intelligence/

[6] Wikipedia. Machine learning. https://en.wikipedia.org/wiki/Machine_learning

[7] ObjectiveMind.ai. The Math Behind the Machine: AI Fundamentals Explained. June 2025. https://www.objectivemind.ai/the-math-behind-the-machine-ai-fundamentals-explained

[8] Relief. The Math Behind Machine Learning Linear Algebra Calculus and Probability. Mar 3, 2025, Medium. https://medium.com/@preetikapuria587/the-math-behind-machine-learning-linear-algebra-calculus-and-probability-91f1e0d1350e

[9] Coursera. Mathematics for Machine Learning and Data Science Specialization. https://www.coursera.org/specializations/mathematics-for-machine-learning-and-data-science

[10] {igebra.ai}. A Data & AI Research. The Power of Math and AI: Synergies for the Future. https://www.igebra.ai/blog/the-power-of-math-and-ai-synergies-for-the-future/

[11] Richmond Alake. A Data Scientist's Guide to Gradient Descent and Backpropagation Algorithms. NVidia Developers Technical Blog. Feb 09, 2022. https://developer.nvidia.com/blog/a-data-scientists-guide-to-gradient-descent-and-backpropagation-algorithms/

[12] GeeksforGeeks. Optimization Rule in Deep Neural Networks. https://www.geeksforgeeks.org/deep-learning/optimization-rule-in-deep-neural-networks/

[13] Mbali Kalirane. Gradient Descent vs. Backpropagation: What's the Difference? Analytics Vidhya. https://www.analyticsvidhya.com/blog/2023/01/gradient-descent-vs-backpropagation-whats-the-difference/

[14] Massimiliano Ferrara. Explainable artificial intelligence and mathematics: What lies behind? Let us focus on this new research field. Eur. Math. Soc. Mag. 135 (2025), pp. 39–44, DOI 10.4171/MAG/235

[15] IBM. What Is Machine Learning (ML)? https://www.ibm.com/think/topics/machine-learning
[16] Oleksii Malyshev. Math at the Heart of CNN: Part 2. Svitla Systems, June 21, 2024, https://svitla.com/blog/math-at-the-heart-of-cnn/
[17] Lamiae Hana. Classical ML vs. Deep Learning: What is the difference between Classical ML & Deep Learning? Medium. https://lamiae-hana.medium.com/classical-ml-vs-deep-learning-f8e28a52132d
[18] Michael Bronstein, Taco Cohen, and Petar Veličković (2023). Towards Geometric Deep Learning. The Gradient. 18 Feb, 2023. https://thegradient.pub/towards-geometric-deep-learning/
[19] Siddharth Bhalsod. Geometric Deep Learning: An In-Depth Exploration of Principles, Applications, and Future Directions. DEV Community. Oct 27, 2024. https://dev.to/siddharthbhalsod/geometric-deep-learning-an-in-depth-exploration-of-principles-applications-and-future-directions-kn6
[20] CAIS. Recurrent Neural Networks. https://caisplusplus.usc.edu/curriculum/neural-network-flavors/recurrent-neural-networks
[21] Wikipedia. Attention (machine learning). https://en.wikipedia.org/wiki/Attention_(machine_learning)
[22] IBM. What is an attention mechanism? https://www.ibm.com/think/topics/attention-mechanism
[23] Muhammad Raza. LLMs vs. SLMs: The Differences in Large & Small Language Models. Splunk. February 17, 2025. https://www.splunk.com/en_us/blog/learn/language-models-slm-vs-llm.html
[24] Tomasz Jażdżewski (2025). LLMs vs classic Machine Learning. VirtusLab. Apr 7, 2025. https://virtuslab.com/blog/data/llm-vs-classic-ml/
[25] Cloudian. 6 Types of AI Workloads, Challenges & Critical Best Practices. https://cloudian.com/guides/ai-infrastructure/6-types-of-ai-workloads-challenges-and-critical-best-practices/
[26] Code Upscale. LLMs vs Traditional ML: What's the Real Difference? Apr 8, 2025. https://codeupscale.com/articles/llms-vs-traditional-ml
[27] Okuno, A., & Yano, K. (2023). A Generalization Gap Estimation for Overparameterized Models via the Langevin Functional Variance. Journal of Computational and Graphical Statistics, 32(4), 1287–1295. https://doi.org/10.1080/10618600.2023.2197488
[28] Stephan Wojtowytsch. Generalization. University of Pittsburgh. https://sites.google.com/site/stephanwojtowytsch/teaching/689-deep-learning/generalization
[29] Jingling Li, Yanchao Sun, Jiahao Su, Taiji Suzuki, Furong Huang. Understanding Generalization in Deep Learning via Tensor Methods. Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics. PMLR 108:504-515, 2020.
[30] Wu, Lei, Zhanxing Zhu and E Weinan. Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes. ArXiv abs/1706.10239 (2017): n. pag.
[31] Trisha Pillay. New AI Architectures To Watch Out For. EM360Tech. 06 August 2025. https://em360tech.com/tech-articles/new-ai-architectures-watch-out
[32] Hamon, R., Junklewitz, H., Sanchez, I. Robustness and Explainability of Artificial Intelligence - From technical to policy solutions, EUR 30040, Publications Office of the European Union, Luxembourg, Luxembourg, 2020, ISBN 978-92-79-14660-5 (online), doi:10.2760/57493 (online), JRC119336.
[33] MATLAB. Interpretability. MATLAB & Simulink – MathWorks. https://www.mathworks.com/discovery/interpretability.html
[34] Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Third Edition, Publisher : Christoph Molnar, March 12, 2025, ISBN-10 : 3911578032, ISBN-13: 978-3911578035
[35] Management Solutions. Interpretability techniques: state of the art - Explainable artificial intelligence (XAI). https://www.managementsolutions.com/sites/default/files/minisite/static/22959b0f-b3da-47c8-9d5c-80ec3216552b/iax/pdf/explainable-artificial-intelligence-en-04.pdf
[36] V Devansh. How Google and Stanford made AI more Interpretable with a 20 year old Technique Why AI Interpretability Has Hit Its First Fundamental Wall. Medium. https://machine-learning-made-simple.medium.com/breaking-down-db-ksvd-scalable-alternating-optimization-for-disentangling-high-dimensional-87d3e2237de4
[37] Wikipedia. Topological deep learning. https://en.wikipedia.org/wiki/Topological_deep_learning
[38] Venkat Manish. Topological Deep Learning: Advancing AI Beyond Conventional Limits. Medium. Feb 1, 2025. https://medium.com/@thatikondamanish/topological-deep-learning-advancing-ai-beyond-conventional-limits-a6be9799759d
[39] Mintu Debnath, Vijay Kumar Salvia, Alighazi Siddiqui, Khalid Ali Qidwai, P. Durga Devi, S. Bavankumar. Mathematical Foundations of

Explainable AI: A Framework based on Topological Data Analysis. Communications on Applied Nonlinear Analysis. Vol. 32, No.9s (2025), DOI: https://doi.org/10.52783/cana.v32.4650

[40] Sethu Iyer. Category Theory in Deep Learning: A New Lens for Abstraction, Composition, and the Nature of Intelligence. Medium. Jan 13, 2025. https://medium.com/@sethuiyer/category-theory-in-deep-learning-a-new-lens-for-abstraction-composition-and-the-nature-of-2806963c677f

[41] AWS. Deep Learning vs Machine Learning - Difference Between Data Technologies. Amazon Web Services, Inc. 2025. https://aws.amazon.com/compare/the-difference-between-machine-learning-and-deep-learning/

[42] Boles. AI leaps from math dunce to whiz. Harvard Gazette. July 23, 2025. https://news.harvard.edu/gazette/story/2025/07/ai-leaps-from-math-dunce-to-whiz/

[43] Whitney Clavin. AI Program Plays the Long Game to Solve Decades-Old Math Problems. www.caltech.edu. Feb 13, 2025. https://www.caltech.edu/about/news/ai-program-plays-the-long-game-to-solve-decades-old-math-problems

[44] AI for Mathematics and Theoretical Computer Science. Simons Institute for the Theory of Computing and SLMath Joint Workshop. 2025. https://simons.berkeley.edu/workshops/simons-institute-theory-computing-slmath-joint-workshop-ai-mathematics-theoretical.