InCred Equities

India

Not Rated

Technology - Others

DeepSeek pops the AI bubble

- Trained at US$5.5m, DeepSeek uses **93% less power** vs. GPT-4 (US$78.3m) & 97% less than Gemini Ultra (US$191.4m) in giving comparable performance.
- API **costs slashed by 80-90%**: Input tokens at US$0.55/m vs. OpenAI's US$15/m redefine AI affordability and run-time power consumption.
- Innovative **reinforcement learning and MoE techniques** set new benchmarks for model efficiency.

DeepSeek's R1 model shocks the markets

On **23 Jan 2025, China-based artificial intelligence or AI startup DeepSeek released** its open-source R1 reasoning generative AI (GenAI) model. News about R1 quickly spread, and by the start of stock trading on 27 Jan 2025, the market capitalization of many major technology companies with a large AI footprint had fallen drastically. **NVIDIA, a dominant force in AI computing, saw an 18% drop** in its stock price within 10 days. **Microsoft, which heavily invested in AI through Azure, fell 7.5%. Broadcom and Siemens Energy, both key players in AI infrastructure, also took double-digit hits**. The market reaction signaled a seismic shift—DeepSeek had just disrupted the AI playing field.

The power-efficient large language model (LLM) revolution

DeepSeek-R1 shatters the myth that more powerful AI models must come with exponentially higher energy costs. With a **training cost of just US$5.5m**, it consumes **93% less power than GPT-4 (US$78.3m) and 97% less than Gemini Ultra (US$191.4m)—** yet delivers **comparable performance**. This efficiency isn't just in training; it extends to run-time usage. **DeepSeek-R1 processes input tokens at just US$0.55/m**, a staggering **97% cheaper than OpenAI's US$15/m**, slashing run-time power consumption. By proving that scaling LLMs can be both **cost-effective and energy-efficient**, DeepSeek-R1 marks a breakthrough in sustainable AI development.

The cost advantage that changes everything

DeepSeek's R1 model wasn't just a technological breakthrough, it was an economic one. While initial reports suggested remarkably **low US$5.5m training costs**, this figure excluded crucial expenses like hardware, salaries, and R&D investments. Despite the uncertainty around its true development costs, one fact remained clear: **DeepSeek had built a cutting-edge AI model at a fraction of what industry giants like OpenAI and Google typically spend**. But the real shock came with its API pricing. **DeepSeek R1's token costs are over 90% cheaper than OpenAI's**, with input tokens priced at just US$0.55/m and output tokens at US$2.19/m—compared to OpenAI's o1 model, which charges US$15/m and US$60/m, respectively. **This drastic cost advantage positioned DeepSeek as an immediate disruptor**, challenging the premium pricing of existing AI services and forcing the industry to rethink its monetization strategies.

Unlike OpenAI, DeepSeek is actually 'Open AI'

DeepSeek R1 stands out not just for its performance and pricing but also for its openness. Unlike most competitors, **DeepSeek has made R1's model weights and training methodologies freely available** on platforms like Hugging Face and GitHub. However, by Open Source Initiative (OSI) standards, R1 still falls short of being truly open source as its original training code and dataset remain undisclosed. Despite this, the AI community has embraced R1's transparency, with Hugging Face launching an Open-R1 initiative to reconstruct the missing training pipeline and push the model towards a full open-source status. This move signals a broader industry shift—one where AI innovation is increasingly driven by open collaboration rather than corporate secrecy. Whether DeepSeek embraces full openness or not, **R1 has already ignited a movement that could redefine the future of AI development.**

Research Analyst(s)

Shubham DALIA
T (91) 02241611544
E shubham.dalia@incredresearch.com

Understanding AI ML

Machine Learning (ML):

Machine Learning (ML) is a specialized branch of AI that enables computers to learn patterns from data without being explicitly programmed. Instead of being given step-by-step instructions, ML models recognize trends in large datasets and use that knowledge to make predictions or decisions

Just imagine you want a computer to recognize spam e-mails. **Instead of programming it with a list of spam words, you give it thousands of examples of spam and non-spam e-mails**. Over time, the model learns the patterns—such as common phrases, sender behaviour, and attachments—and starts filtering e-mails automatically.

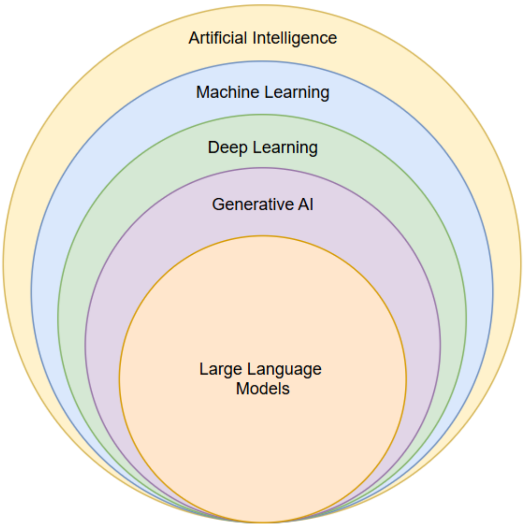
Real-world applications ›

Fraud detection: Banks use ML to flag unusual transactions, such as a sudden large purchase in another country, which might indicate credit card fraud.

Personalized ads: Google and Facebook track your browsing behaviour to show advertisements that match your interests.

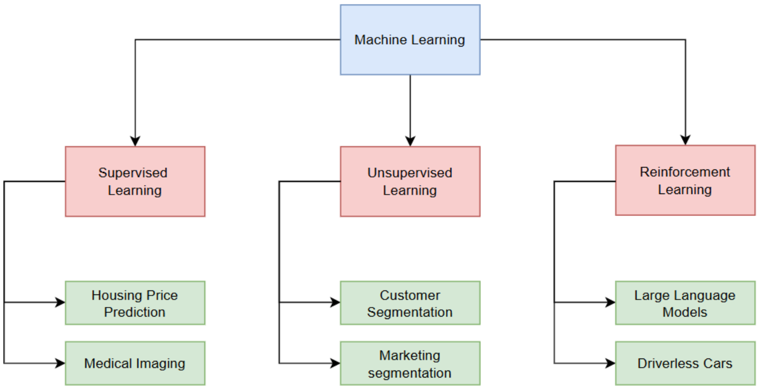
Medical diagnosis: AI-powered systems like IBM Watson analyze medical records to help doctors diagnose diseases.

Figure 1: Diagram representing the hierarchy of systems



SOURCE: INCRED RESEARCH

Figure 2: Types of ML (broadly speaking)



SOURCE: INCRED RESEARCH

Branches of Machine Learning

There are three most used categories of Machine Learning:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

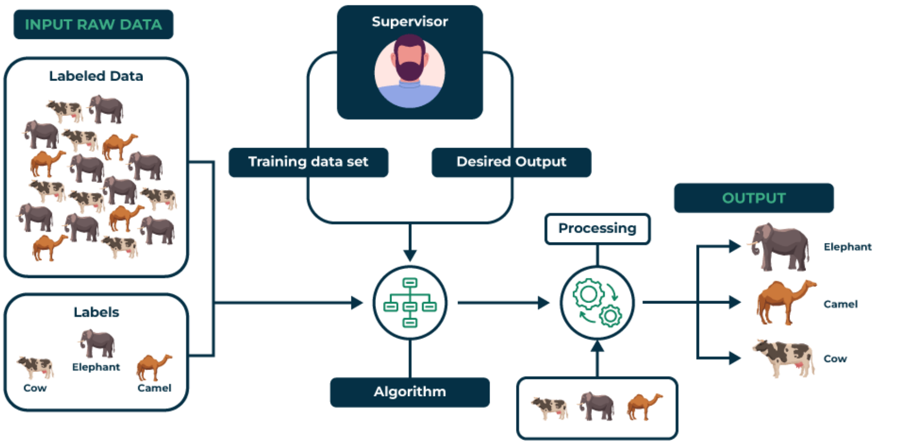
Supervised learning is a type of machine learning where algorithms are trained using labeled datasets to identify patterns and make predictions. This approach enables the development of sophisticated models capable of delivering accurate forecasts, making it widely applicable across industries such as healthcare, marketing, and finance.

In supervised learning**, the dataset consists of input features paired with corresponding correct outputs, allowing the algorithm to learn from past examples**. By analyzing numerous such training instances, the model establishes relationships between inputs and outputs, enabling it to make informed predictions when presented with new data.

For example, consider training a model to recognize different tree species. The dataset would contain various tree images along with their respective species' labels. The algorithm analyzes these labeled examples to identify distinguishing characteristics. Once trained, the model can be tested by providing an image of a tree and asking it to predict the species. If it makes an incorrect prediction, further training with additional data can refine its accuracy and reduce errors.

After sufficient training and testing, the model becomes capable of making predictions on unseen data, leveraging the patterns it has learned from previous examples.

Figure 3: Supervised learning



SOURCE: GEEKSFORGEEKS, INCRED RESEARCH

Unsupervised learning in artificial intelligence is a type of machine learning that learns from data without human supervision

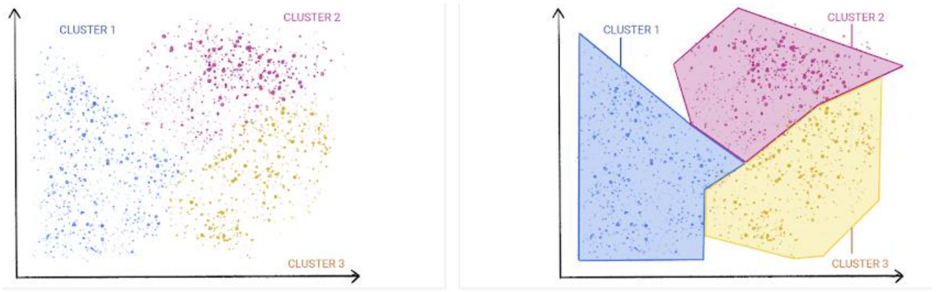
As the name suggests, unsupervised learning uses self-learning algorithms—**they learn without any labels or prior training. Instead**, the model is given raw, unlabelled data and has to infer its own rules and structure the information based on similarities, differences, and patterns without explicit instructions on how to work with each piece of data.

Unsupervised learning algorithms are better suited for more complex processing tasks, such as organizing large datasets into clusters. They are useful for identifying previously undetected patterns in data and can help identify features useful for categorizing data.

Imagine that you have a large dataset about weather. An unsupervised learning algorithm will go through the data and identify patterns in the data points. For instance, it might group data by temperature or similar weather patterns.

While the algorithm itself does not understand these patterns based on any previous information you provided, you can then go through the data groupings and attempt to classify them based on your understanding of the dataset. For instance, you might recognize that the different temperature groups represent all four seasons or that the weather patterns are separated into different types of weather, such as rain, sleet, or snow.

Figure 4: Unsupervised learning



SOURCE: GOOGLE, INCRED RESEARCH

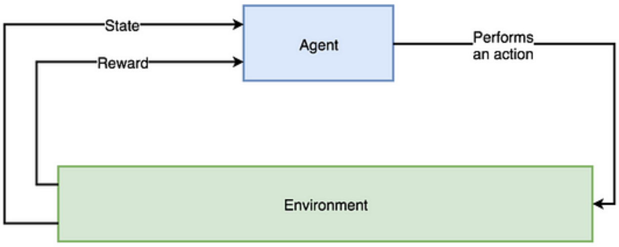
Reinforcement Learning is a branch of machine learning that focuses on developing autonomous agents capable of interacting with their environment and making decisions.

Unlike traditional machine learning models, an RL agent continuously learns by receiving information—known as the state—from its environment and taking actions in response. Each action results in a reward, which can be positive or negative, serving as feedback to guide the agent's learning process.

The training process in reinforcement learning follows a trial-and-error approach. The agent repeatedly explores different scenarios, refining its decision-making by adjusting its parameters based on the rewards or penalties received. Optimization strategies vary and can be based on methods such as value functions, policy gradients, or environment modeling.

Reinforcement learning has a wide range of applications, many of which are at the forefront of AI innovation. It is commonly used in robotics, autonomous systems, game playing, and complex decision-making tasks that require adaptive learning.

Figure 5: Supervised learning



SOURCE: INCRED RESEARCH

Neural Networks

Heart of modern machine learning ➤

Neural networks are the foundation of modern machine learning, particularly deep learning. Modeled after the human brain, they consist of multiple layers of interconnected nodes (neurons), each processing information and passing it to the next layer.

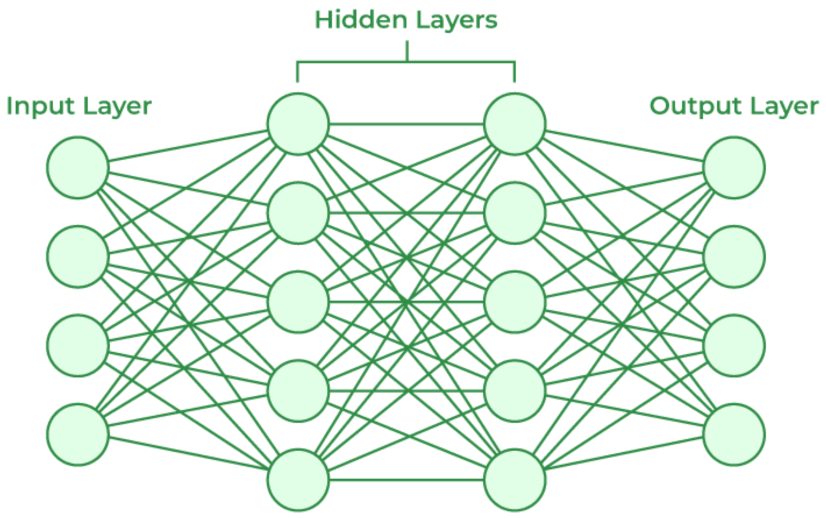
Structure of a neural network

Input layer: Receives raw data, such as pixels in an image or words in a sentence.

Hidden layers: Perform transformations on the input using mathematical operations like weighted sums and activation functions. The presence of multiple hidden layers forms a deep neural network, enhancing its ability to recognize complex patterns.

Output layer: Generates the final prediction, such as identifying whether an image contains a dog or a cat.

Figure 6: Structure of a basic neural network



SOURCE: GEEKSFORGEEKS, INCRED RESEARCH

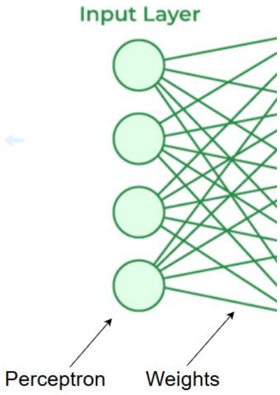
Key concepts in neural networks ➤

Perceptrons: The simplest unit of a neural network, performing a weighted sum of inputs followed by an activation function. These form the building blocks of more advanced networks.

Weights and biases: Weights determine the influence of connections between neurons, while biases allow the model to adjust outputs. These parameters are fine-tuned during training to minimize prediction errors.

Activation functions: These introduce non-linearity into the network, enabling it to capture complex relationships in data. Common activation functions include rectified linear unit, sigmoid, and tanh. Without them, neural networks would behave like simple linear models.

Figure 7: Perceptron and weights



SOURCE: GEEKSFORGEEKS, INCRED RESEARCH

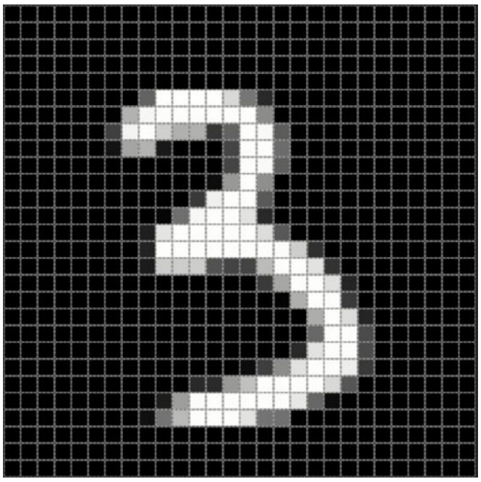
Structure in depth for basic image recognition ➤

A neural network is a system of interconnected neurons that process numerical information. Each neuron holds a value, known as its activation, which falls within a range—typically between 0 and 1. The network consists of multiple layers, including an input layer, hidden layers, and an output layer.

- **Input layer:** Holds the raw data, such as pixel values in an image. For instance, a grayscale image of 24×24 pixels is represented by 576 neurons, each storing a brightness value between 0 (black) and 1 (white).
- **Hidden layers:** These intermediate layers process and transform the input data to extract patterns. They help in recognizing basic components like edges, loops, or other visual features before passing the processed data forward.
- **Output layer:** Produces the final prediction. In a digit classification task, this layer consists of 10 neurons, each representing a possible digit (0-9). The activation of each neuron indicates the confidence level of the network's prediction.

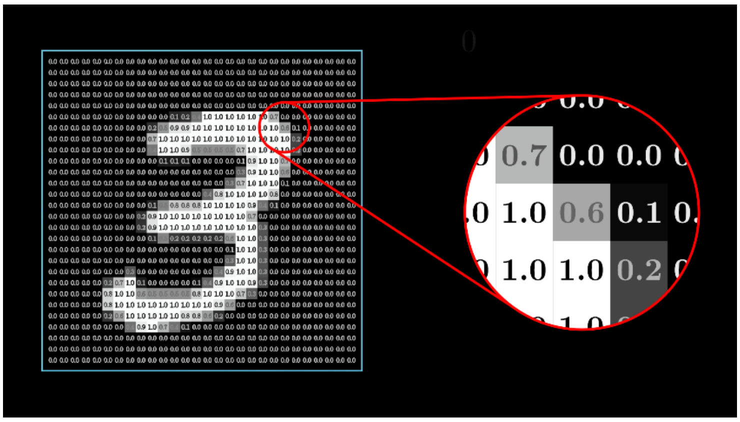
The hidden layers are crucial because they allow the network to detect hierarchical patterns—starting with simple edges, then forming shapes, and finally recognizing complex structures like digits or objects.

Figure 8: Pixel representation of 3



SOURCE: MNIST DATABASE, INCRED RESEARCH

Figure 9: Pixel representation of 3 with weights to perceptron



SOURCE: 3BLUE1BROWN,INCRED RESEARCH

How information passes between layers

The key mechanism driving a neural network is the weighted connections between neurons. Each neuron in one layer is connected to neurons in the next layer through weights, which determine how much influence one neuron has on another. Mathematically, each neuron's activation is calculated using a weighted sum of the activations from the previous layer: **∑(wi·ai)**

= *w*1*a*1 + *w*2*a*2 + *w*3*a*3 + *w*4*a*4 + ⋯ + *wnan*

Where:
wi represents the weight of the connection,

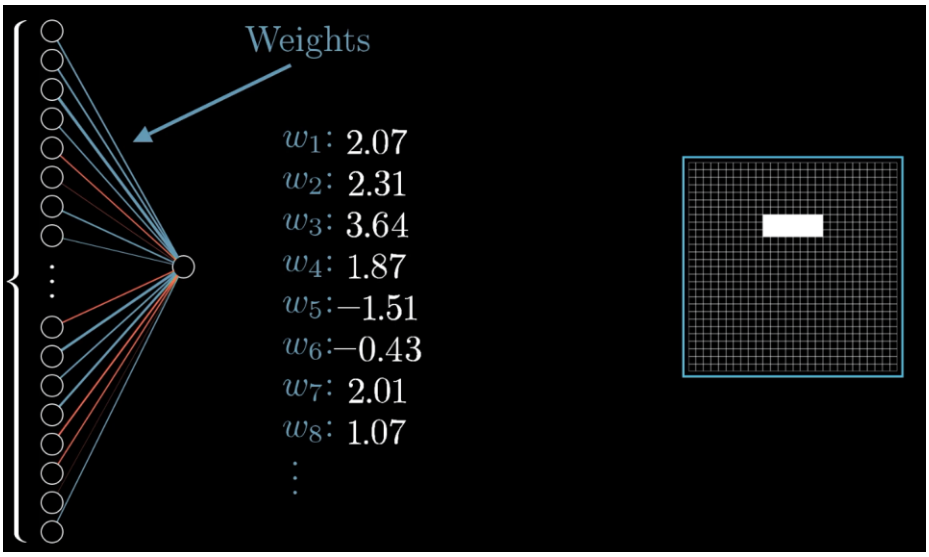
ai is the activation of the neuron from the previous layer.

In simple terms, a positive weight strengthens the connection, meaning that if the previous neuron is activated, the current neuron is also likely to activate. A negative weight weakens the connection, reducing the likelihood of activation.

To ensure that activations remain within a desired range (between 0 and 1), the weighted sum is passed through an activation function like the sigmoid function, which "compresses" the values into the range [0,1]:

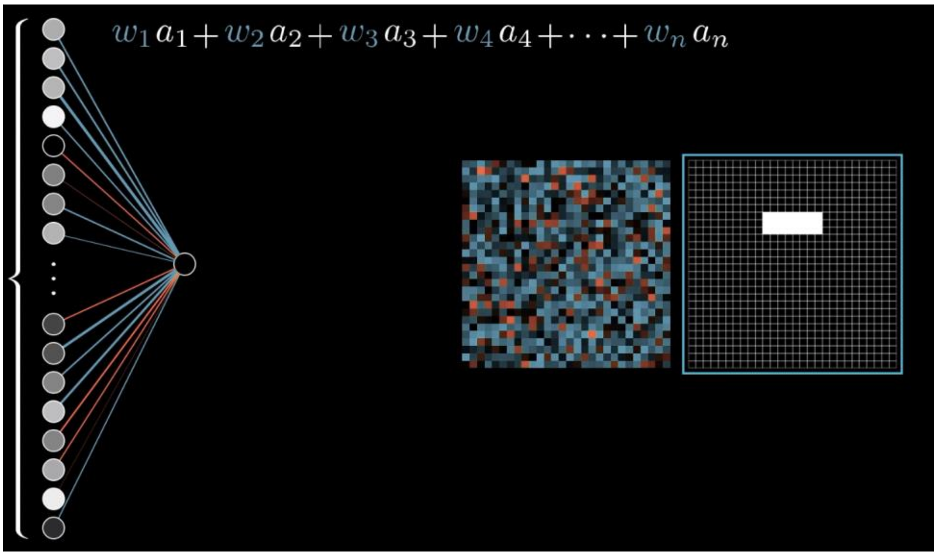
This transformation allows the network to make smooth, probabilistic decisions rather than binary ones.

Figure 10: How information is passed -I



SOURCE: 3BLUE1BROWN, INCRED RESEARCH

Figure 11: How information is passed -2



SOURCE: 3BLUE1BROWN, INCRED RESEARCH

The complexity of a neural network ➤

Each neuron in a hidden layer has a unique set of weights for its connections to the previous layer. Additionally, a bias value is added to the weighted sum before applying the activation function, allowing neurons to adjust outputs independently of the input.

For a simple network with:

- 576 input neurons (one for each pixel in a 24×24 image),
- 16 neurons in the first hidden layer
- 16 neurons in the second hidden layer
- 10 output neurons.

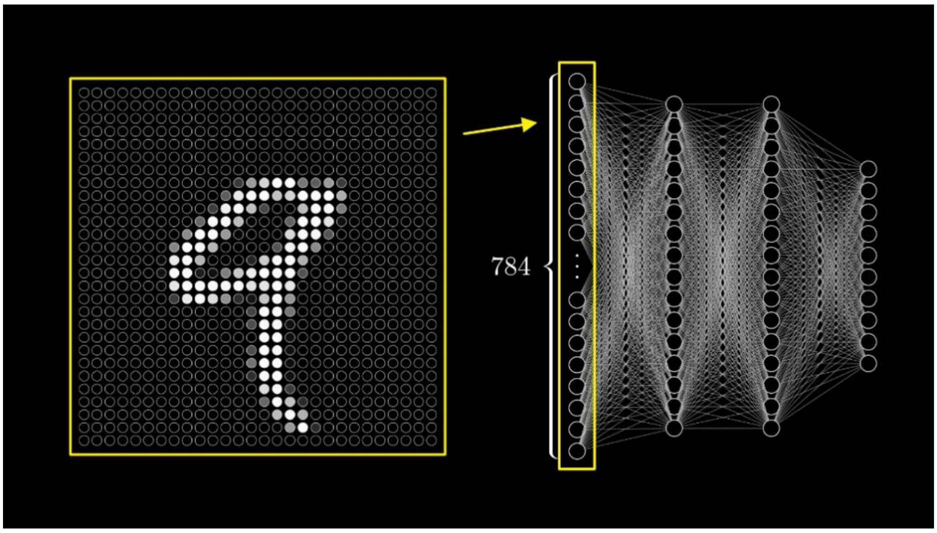
The total number of parameters (weights and biases) that can be adjusted during training exceeds 9,600. These parameters act as "knobs and dials" that fine-tune the network’s behaviour.

A neural network as a function ➤

At its core, a neural network is just a mathematical function—**one that transforms input values into meaningful predictions**. While it may seem complex due to the large number of parameters and layers, **its purpose remains simple: mapping inputs to outputs in an optimal way through learned patterns**.

The learning process involves adjusting weights and biases through optimization techniques like gradient descent, refining the network over time to improve accuracy. This iterative process enables neural networks to recognize patterns, make predictions, and generalize to new data effectively.

Figure 12: Pixel representation of 9

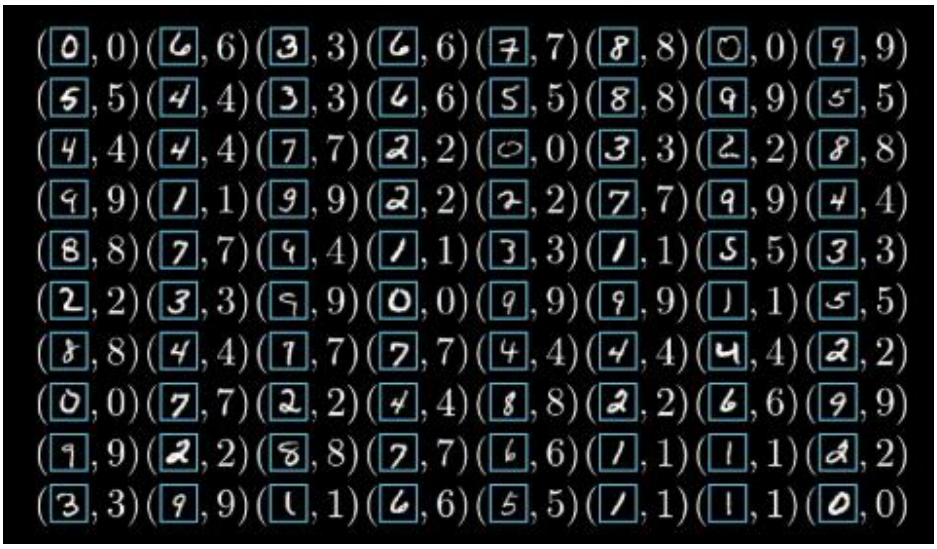


SOURCE: 3BLUE1BROWN, INCRED RESEARCH

How a neural network learns

Machine learning distinguishes itself from traditional programming paradigms by **not requiring explicit instructions to perform a given task**. Instead of manually crafting an algorithm to recognize handwritten digits, a neural network is structured to learn from labelled data. This learning process involves the adjustment of numerous weights and biases—parameters that dictate the network's ability to generalize beyond its training set.

Figure 13: Training dataset



SOURCE: 3BLUE1BROWN, INCRED RESEARCH

Training data and generalization ➤

The training process is dependent on a dataset comprising labelled images, commonly referred to as the "training data." The underlying assumption is that through iterative exposure to these examples, the network will develop feature representations that extend beyond the given dataset. To assess this capability, a separate validation dataset is utilized to measure the network's classification accuracy on previously unseen samples.

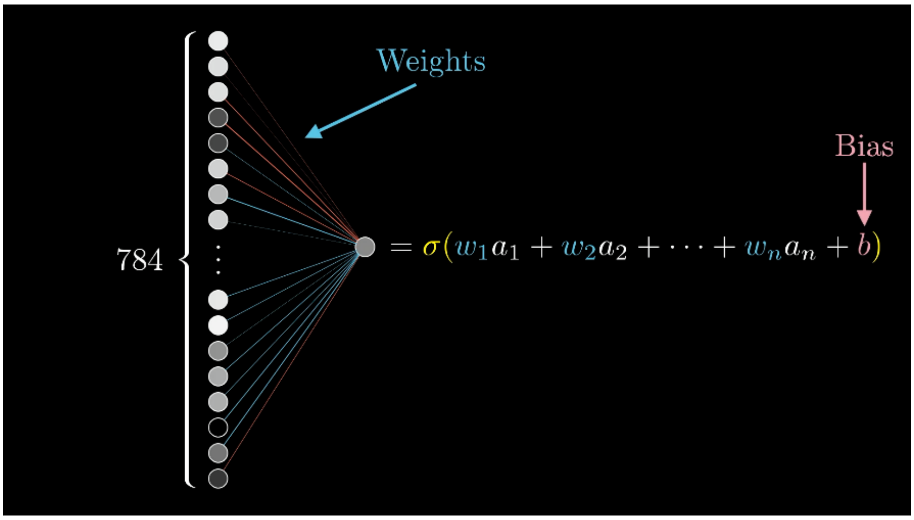
The role of weights and biases

At the core of neural network learning is the modulation of weights and biases, which define the strength of interconnections between neurons. Initially, these parameters are randomly assigned, resulting in poor performance. Through successive training iterations, these values are systematically refined to optimize performance.

Cost Function: Quantifying performance

The cost function is instrumental in evaluating the network's performance by quantifying the discrepancy between its predicted outputs and the expected results. For instance, in the context of digit classification, an ideal network would activate a single output neuron corresponding to the correct digit while suppressing activations in others. The deviation from this ideal output is computed using the sum of squared differences between predicted activations and the target values.

Figure 14: Weights and biases

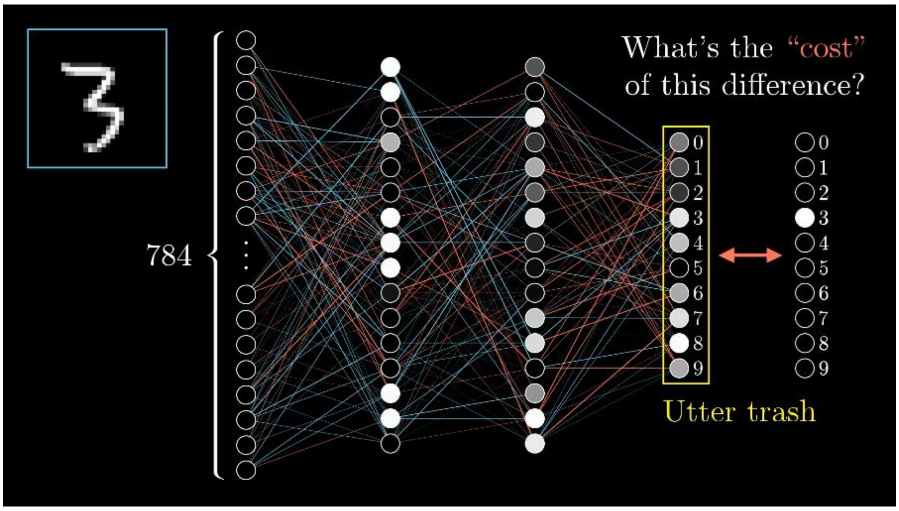


SOURCE:3BLUE1BROWN INCRED RESEARCH

Cost function over an entire dataset

While evaluating performance on a single example is insightful, an effective learning process must minimize the average cost over all training samples. Given the extensive parameter space—often consisting of tens of thousands of weights and biases—this results in a highly complex function that must be optimized iteratively.

Figure 15: Weights and biases



SOURCE:3BLUE1BROWN INCRED RESEARCH

Backpropagation: The learning mechanism of neural nets ➤

Backpropagation, introduced in 1986 by David Rumelhart, Geoffrey Hinton, and Ronald Williams, revolutionized the training of artificial neural networks. It became the standard algorithm for optimizing **Multi-Layer Perceptrons (MLPs) and was later adopted for more complex architectures like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs).** Backpropagation enables neural networks to efficiently adjust their parameters—weights and biases—even when dealing with models containing thousands or billions of parameters.

Understanding backpropagation ➤

Backpropagation is the reverse process of forward propagation in a neural network. While forward propagation moves input data through the network to produce an output, backpropagation traces errors backward from the output layer to the input layer to update the network's parameters.

Forward Propagation: Input → Hidden Layers → Output

Backpropagation: Output Error → Hidden Layers → Input

During forward propagation, the network makes a prediction, which is compared against the correct label to calculate an error (loss). Backpropagation then determines how this error can be minimized by adjusting the weights and biases across the network.

How backpropagation works ➤

Compute the error:

The network's prediction is compared with the actual label using a loss function (e.g., Mean Squared Error for regression or Cross-Entropy for classification).

Propagate the error backward:

The error is distributed backward through the network using the chain rule of calculus, determining how much each weight contributed to the total error.

Update weights and biases:

Each weight is adjusted using Gradient Descent, an optimization algorithm that nudges weights in the direction that reduces error the most.

The learning rate controls how big these adjustments are—too high can overshoot the optimal values, while too low can slow down learning.

Why backpropagation is essential ➤

Without backpropagation, training deep neural networks would be impractical. It automates the learning process by efficiently finding the optimal weights that minimize errors and improve prediction accuracy. This method remains the backbone of modern AI systems, powering everything from image recognition and language models to financial forecasting and self-driving cars.

Large Language Models (LLMs)

How do they work

Large Language Models (LLMs) like GPT-3 and GPT-4 have transformed artificial intelligence by enabling machines to generate human-like text, answer complex questions, and even write code. Unlike traditional software, which follows step-by-step instructions from human programmers, **LLMs are trained using vast amounts of text data and operate using neural networks**. This makes their decision-making process difficult to interpret fully, but researchers have made significant progress in understanding their mechanisms.

How LLMs represent words

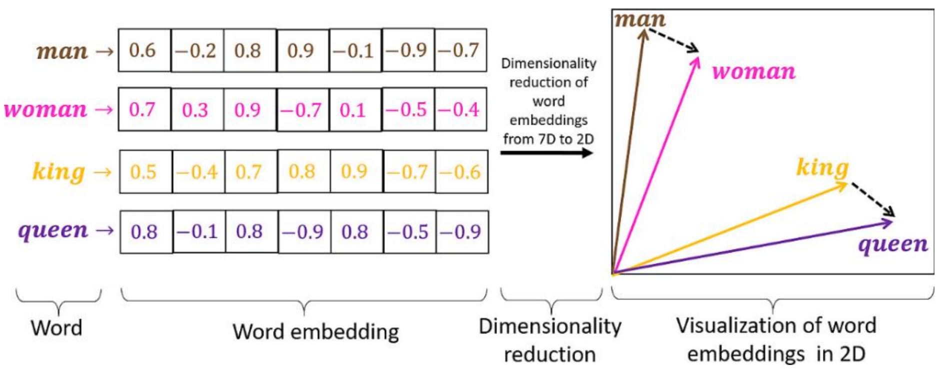
A fundamental aspect of LLMs is how they represent words. Instead of storing words as sequences of letters (e.g., "C-A-T" for cat), LLMs use word vectors—long lists of numbers that capture the relationships between words.

For example, words with similar meanings, such as "cat," "dog," and "kitten," are placed close to each other in a multi-dimensional vector space. This enables LLMs to recognize relationships between words and reason about them mathematically.

We can represent this using a vector notation, example:

- Washington DC is at [38.9, 77]
- New York is at [40.7, 74]
- London is at [51.5, 0.1]
- Paris is at [48.9, -2.4]

Figure 16: 2D representation of word embeddings



man →	0.6	−0.2	0.8	0.9	−0.1	−0.9	−0.7
woman →	0.7	0.3	0.9	−0.7	0.1	−0.5	−0.4
king →	0.5	−0.4	0.7	0.8	0.9	−0.7	−0.6
queen →	0.8	−0.1	0.8	−0.9	0.8	−0.5	−0.9

Dimensionality reduction of word embeddings from 7D to 2D
man
woman
king
queen
Word
Word embedding
Dimensionality reduction
Visualization of word embeddings in 2D

SOURCE:AIRBYTE,INCRED RESEARCH

One key breakthrough in word vectors came from Google's word2vec project in 2013, which trained neural networks to identify words that often appear in similar contexts. This allowed for fascinating operations, such as:

- Biggest - big + small = smallest
- Paris - France + Germany = Berlin

However, a major limitation of simple word vectors is that they don't account for multiple meanings of a word. For example, the word "bank" could refer to a financial institution or the side of a river. LLMs solve this by dynamically adjusting word vectors based on context, ensuring that "bank" in "John went to the bank" is interpreted differently than in "The river bank was eroding."

Transformers: The key innovation behind LLMs

LLMs like GPT-3 and GPT-4 rely on transformers, a neural network architecture introduced in a 2017 paper by Google. Transformers process text efficiently by using self-attention mechanisms, which allow words to "attend" to other words in a sentence and understand their relationships.

For example, **in the sentence "John gave his bank details to the teller," the word "his" needs to be linked to "John" for the model to understand the meaning correctly.** Transformers achieve this by assigning numerical values to relationships between words and adjusting them dynamically across multiple processing layers.

Each transformer consists of multiple attention layers that refine the meaning of words by considering context at different levels. Early layers focus on syntax and word relationships, while deeper layers develop a high-level understanding of sentences, paragraphs, or entire documents.

Figure 17: Transformer architecture

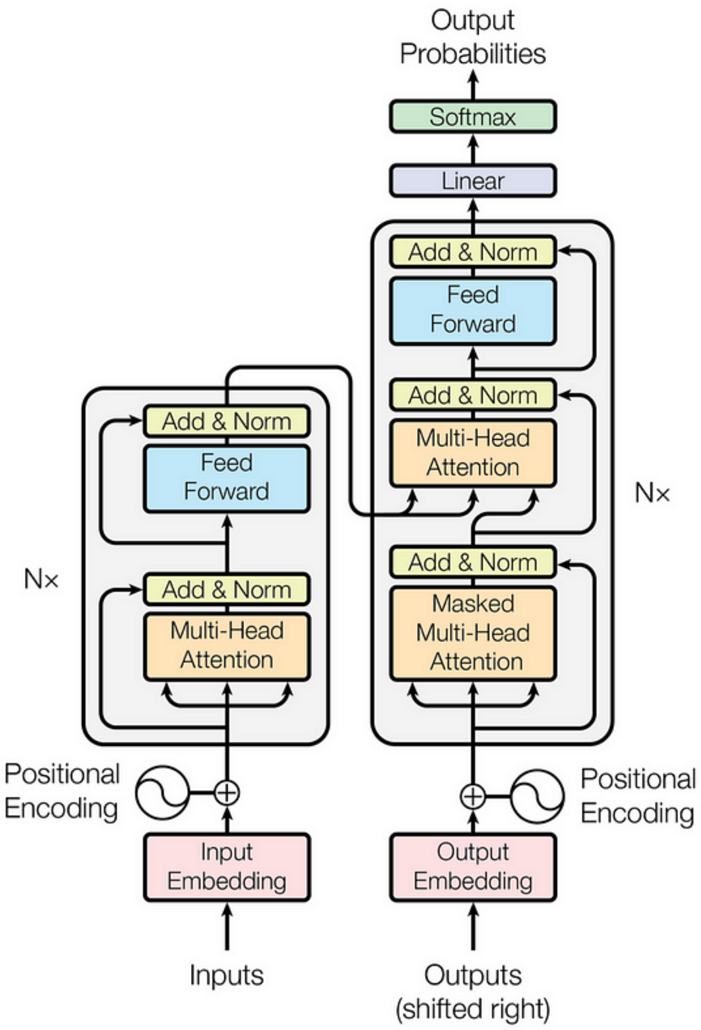


Figure 1: The Transformer - model architecture.

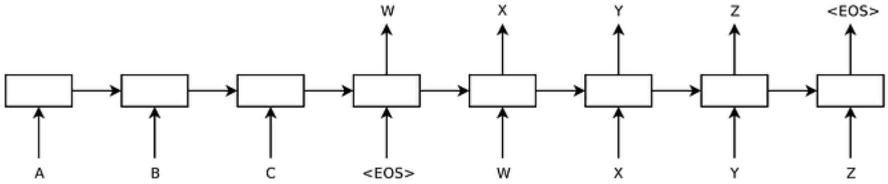
SOURCE:AIRBYTE,INCRED RESEARCH

The sequential nature of RNNs ›

Recurrent Neural Networks (RNNs), particularly those using Long Short-Term Memory (LSTM) units, handle tasks like language translation in a sequential manner. As outlined in the seminal paper ‘Sequence to Sequence Learning with Neural Networks — 2014,’ RNNs process tokens one at a time, allowing the model to capture the order and dependencies inherent in the language.

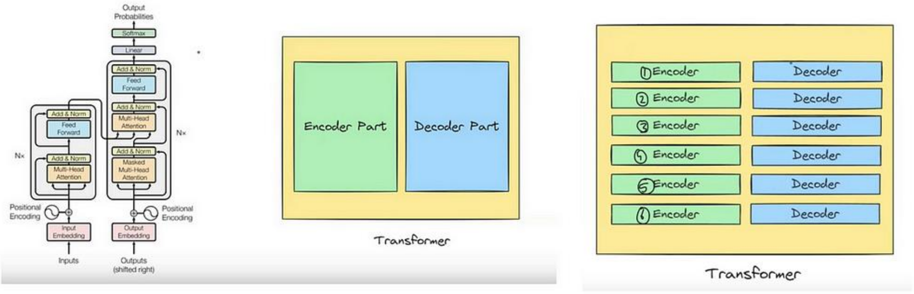
In a typical translation task, an RNN encoder reads the source language (for instance, words A, B, and C in English) sequentially, and the decoder subsequently generates the target language (words such as W, X, Y, Z in French) by predicting one word after another, with <EOS> marking the end of the sequence. **This one-by-one processing, while effective for capturing sequence relationships, inherently limits training speed—especially with large datasets—because each token must be processed in order.** The paper’s approach of using separate LSTMs for encoding and decoding exemplifies how RNNs structure their computations to manage complex translation tasks despite these limitations.

Figure 18: RNN handles sequence-to-sequence task



SOURCE: INCRED RESEARCH

Figure 19: Transformer architecture



SOURCE:CAMPUSX,INCRED RESEARCH

Attention Mechanism: The key to context understanding ➤

Within transformers, attention heads determine which words are most relevant to a given word. For example, in "John gave a drink to Mary," attention mechanisms help the model recognize that "Mary" is the recipient of the drink, rather than "John."

Each layer has multiple attention heads focusing on different linguistic tasks, such as:

- Resolving pronouns (e.g., linking "his" to "John").
- Understanding homonyms (e.g., distinguishing "bank" as a financial institution vs. a riverbank).
- Detecting phrase structures (e.g., recognizing "Joe Biden" as a single entity).

The largest version of GPT-3 has 96 layers with 96 attention heads per layer, meaning that each word undergoes over 9,000 contextual adjustments before the model predicts the next word.

Scaling and Performance: Why bigger models work better ➤

A key reason for the success of LLMs is scale. OpenAI's research shows that model performance improves consistently as the number of parameters, training data, and computing power increase.

For example:

- GPT-1 (2018) had 117m parameters.
- GPT-2 (2019) expanded to 1.5bn parameters.
- GPT-3 (2020) scaled to 175bn parameters.
- GPT-4 (2023) is rumoured to be even larger, although the exact details remain undisclosed.

The sheer scale of training data allows LLMs to develop advanced reasoning, analogy-making, and problem-solving abilities, even though they are only trained to predict words.

ChatGPT architecture

Large Language Models (LLMs) like ChatGPT have transformed natural language processing (NLP) by generating human-like responses across various contexts. Unlike traditional software that follows explicit rules, ChatGPT is built on deep learning principles, leveraging transformer architectures, self-attention mechanisms, and pre-training with fine-tuning to produce coherent and context-aware text.

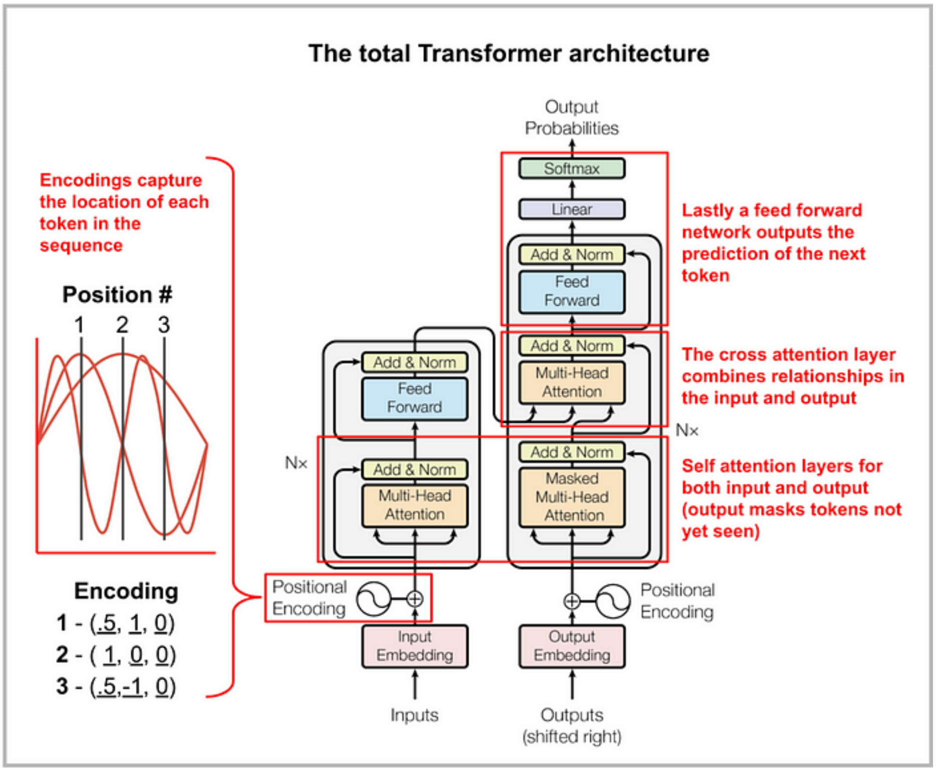
Transformer Architecture: The core of ChatGPT ➤

At the heart of ChatGPT lies the transformer architecture, introduced by Vaswani et al. in 2017. This framework revolutionized NLP by enabling models to process entire sequences of text at once rather than word by word. The transformer consists of two primary components:

- Encoder: Processes input text and extracts meaningful features.
- Decoder: Generates responses based on the processed input.

ChatGPT primarily relies on the decoder-only variant of the transformer, meaning it focuses on generating text by predicting the next token in a sequence while considering the broader context of previous words.

Figure 20: Tokenization



Encodings capture the location of each token in the sequence

Lastly a feed forward network outputs the prediction of the next token

The cross attention layer combines relationships in the input and output

Self attention layers for both input and output (output masks tokens not yet seen)

SOURCE:INCRED RESEARCH

Self-Attention Mechanism: Understanding context ➤

A major breakthrough in transformer models is self-attention, which allows ChatGPT to weigh the importance of different words when generating a response. **Unlike traditional models such as Recurrent Neural Networks (RNNs), which process text sequentially, self-attention enables the model to analyze all words in a sentence simultaneously**, making it highly efficient in capturing long-range dependencies.

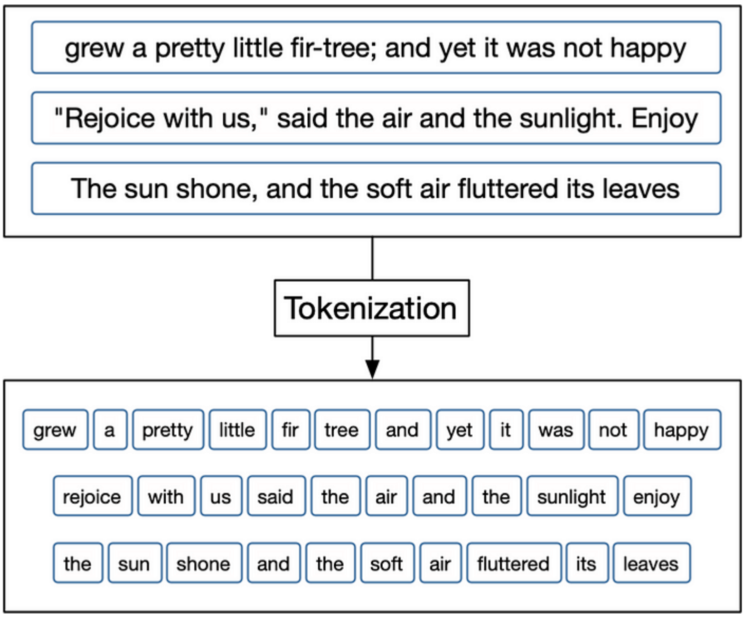
This mechanism assigns **attention scores to words based on their relevance, allowing ChatGPT to identify relationships between words**, even if they are far apart in a sentence. It also helps resolve ambiguities in language, such as distinguishing between "bank" as a financial institution and a “riverbank”. Additionally, self-attention enhances the model’s ability to retain context across multiple sentences in a conversation, making responses more coherent and relevant.

Tokenization: Breaking text into units ➤

Before ChatGPT processes text, it must be tokenized, which means breaking text into smaller components called tokens**. These tokens may represent entire words or sub-words, such as how "play," "playing," and "played" share common roots.**

Tokenization is crucial because it allows the model to efficiently handle words of varying lengths, ensuring that it can process text effectively. It also helps recognize morphemes, the smallest units of meaning in a language, which aids in understanding complex words. By breaking words down into smaller units, tokenization reduces the complexity of large vocabularies, making it easier for the model to learn and represent words as combinations of fundamental components rather than as isolated entities.

Figure 21: Tokenization



grew a pretty little fir-tree; and yet it was not happy

"Rejoice with us," said the air and the sunlight. Enjoy

The sun shone, and the soft air fluttered its leaves

Tokenization
grew a pretty little fir tree and yet it was not happy
rejoice with us said the air and the sunlight enjoy
the sun shone and the soft air fluttered its leaves

SOURCE:SMLTAR,INCRED RESEARCH

Word Embeddings: Capturing semantic meaning ➤

ChatGPT converts tokens into vector embeddings, which are numerical representations that capture the relationships between words. **These embeddings are essential because they position words with similar meanings closer together in a multi-dimensional space.** For example, "dog" and "puppy" would have similar vector representations, reflecting their semantic relationship.

This structure enables the model to generalize knowledge from one word to similar words, improving its overall text understanding. **Additionally, embeddings provide context-awareness, helping the model determine word meanings based on their surrounding context.** This allows ChatGPT to generate responses that are not only relevant but also more natural and contextually accurate.

Layered Processing: Learning complex patterns ➤

ChatGPT consists of multiple layers of self-attention and feed-forward neural networks, each refining the information passed from the previous layer and progressively capturing more abstract linguistic patterns. The lower layers primarily detect simple patterns, such as sentence structure and basic syntax.

As the information moves through the middle layers, the model captures more advanced grammatical rules and sentence formations. The higher layers focus on reasoning, coherence, and long-term dependencies, enabling

ChatGPT to generate responses that feel structured, logical, and aligned with natural human conversation. The depth of these layers is what allows ChatGPT to process and produce text that is sophisticated, contextually relevant, and highly adaptive to different queries.

Pre-training: Learning from massive text data ➤

Before ChatGPT can engage in meaningful conversations, it undergoes pre-training using vast amounts of publicly available text from books, articles, and websites. During this phase, the model learns statistical patterns in language, including grammar, syntax, and factual information.

It also improves sentence completion by predicting missing words in incomplete sentences, enhancing its ability to generate smooth and coherent text. Furthermore, pre-training helps ChatGPT acquire common-sense knowledge by repeatedly analyzing language usage across different contexts. This process is unsupervised, meaning the model does not rely on labelled data but instead learns to predict the next word in a sequence, gradually improving its understanding of the language.

Fine-Tuning: Aligning responses to human expectations ➤

Following pre-training, the model undergoes fine-tuning, where it is further trained on curated datasets with human-reviewed responses. **This stage ensures that the model's outputs are ethical, safe, and aligned with human conversational styles.** Fine-tuning helps reduce biases and filter out inappropriate content, making the model more reliable in diverse interactions.

To refine its responses further, the model is often trained using Reinforcement Learning from Human Feedback (RLHF), where human reviewers evaluate its generated responses and guide future behaviour. This additional training step enhances the accuracy, consistency, and contextual awareness of ChatGPT, making it better suited for real-world applications.

Context Retention: Enabling interactive conversations ➤

One of ChatGPT's key strengths is its ability to remember context within a conversation. Instead of treating each query in isolation, the model considers previous exchanges, allowing for coherent multi-turn interactions.

This enables it to maintain short-term context, retaining information from prior sentences within a conversation, as well as recognize long-term dependencies, where references from multiple exchanges are understood and appropriately responded to. However, ChatGPT does not have true memory and cannot recall past interactions after a conversation ends. Instead, it maintains temporary context within session limits, ensuring that responses remain relevant within a given conversation but do not persist beyond it.

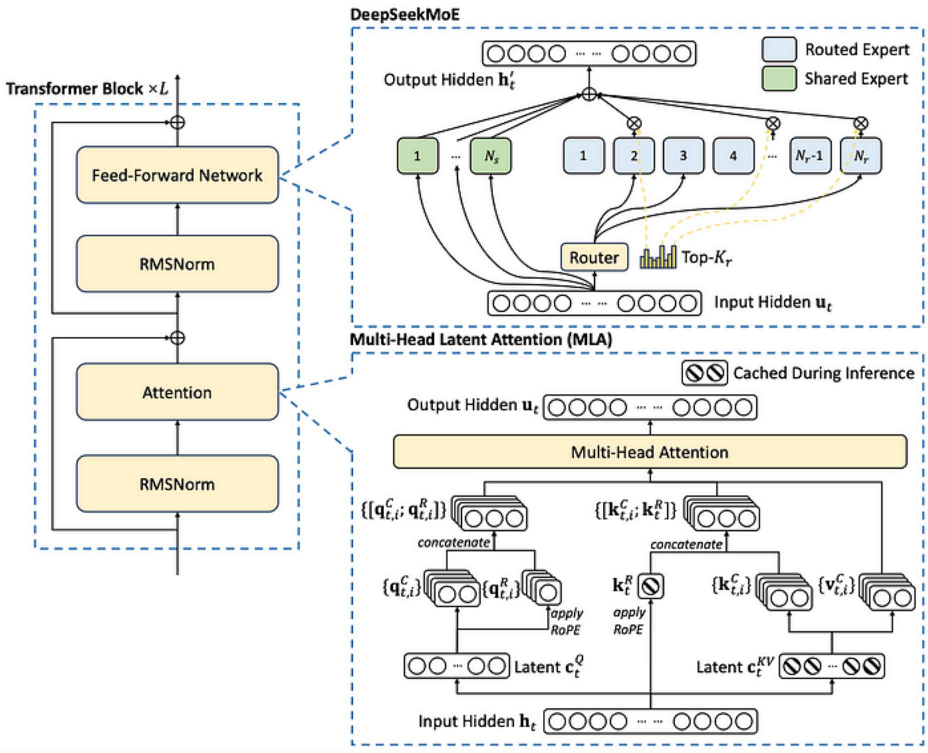
DeepSeek architecture

In the rapidly evolving landscape of large language models (LLMs), DeepSeek-R1 represents a major breakthrough, particularly in mathematical reasoning, coding, and scientific problem-solving.

Unlike traditional LLMs, which rely heavily on Supervised Fine-Tuning (SFT) and Reinforcement Learning with Human Feedback (RLHF), DeepSeek-R1 explores the potential of Reinforcement Learning (RL) as the sole post-training technique.

This innovation builds upon DeepSeek-V3, an advanced Mixture-of-Experts (MoE) model developed by DeepSeek-AI. With 671bn total parameters and 37bn activated parameters per token, **DeepSeek-V3 incorporates architectural advancements such as Multi-Head Latent Attention (MLA), MoE routing optimization, and Multi-Token Prediction (MTP) to achieve efficiency and scalability.** These innovations enable DeepSeek-R1 to push the boundaries of RL-driven LLM training while maintaining state-of-the-art performance.

Figure 22: MoE transformer layer



SOURCE:SMLTAR,DEEPSEEK PAPER,INCRED RESEARCH

The Experiment: DeepSeek-R1-Zero & the RL-only spproach ➤

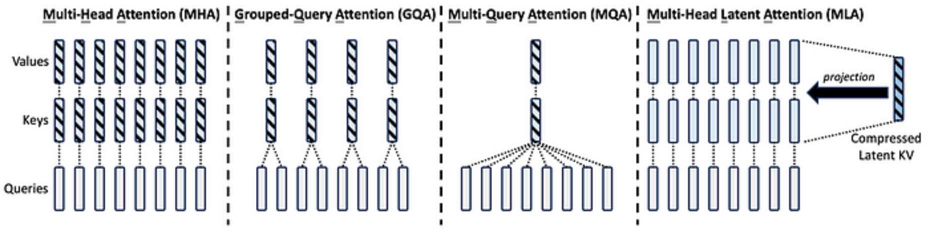
DeepSeek-R1-Zero was designed as an RL-only model, eliminating Supervised Fine-Tuning (SFT) entirely. The key goal was to explore how far Reinforcement Learning alone could drive LLM performance without the need for manually labelled datasets. DeepSeek-R1-Zero was trained using Group Relative Policy Optimization (GRPO), a reinforcement learning algorithm derived from Proximal Policy Optimization (PPO). This approach enhances the model's ability to reason through complex problems while optimizing memory efficiency.

Unlike most LLMs that use neural reward models, **DeepSeek-R1-Zero employs a rule-based reward system, avoiding issues like reward hacking**. The reward model consists of two key components: accuracy rewards, which evaluate whether the response is correct, and format rewards, which enforce structured reasoning using <think>...</think> tags. By avoiding neural reward models, DeepSeek-R1-Zero simplifies the training pipeline and reduces computational costs.

Despite being trained without Supervised Fine-Tuning, DeepSeek-R1-Zero exhibited several notable capabilities. The model demonstrated self-reflection by revisiting and refining previous steps, explored alternative solutions dynamically, and showed signs of self-evolution, where response length steadily increased as the model learned to leverage test-time computation scaling.

However, **despite its impressive reasoning abilities, R1-Zero had significant limitations. The most notable challenges were poor readability, where responses lacked fluency and clarity, and language mixing, where the model struggled to maintain consistency in responses**. To address these shortcomings, DeepSeek AI introduced DeepSeek-R1, incorporating a multi-stage post-training process that combined SFT with RL.

Figure 23: Comparison of MLA with MHA, GQA and MQA


Multi-Head Attention (MHA)
Grouped-Query Attention (GQA)
Multi-Query Attention (MQA)
Multi-Head Latent Attention (MLA)
Values
Keys
Queries
projection
Compressed Latent KV

SOURCE: DEEPSEEK PAPER,INCRED RESEARCH

DeepSeek-R1: A multi-stage post-training approach

Unlike R1-Zero, DeepSeek-R1 follows a structured post-training process that blends Supervised Fine-Tuning (SFT) with Reinforcement Learning to refine both reasoning and readability.

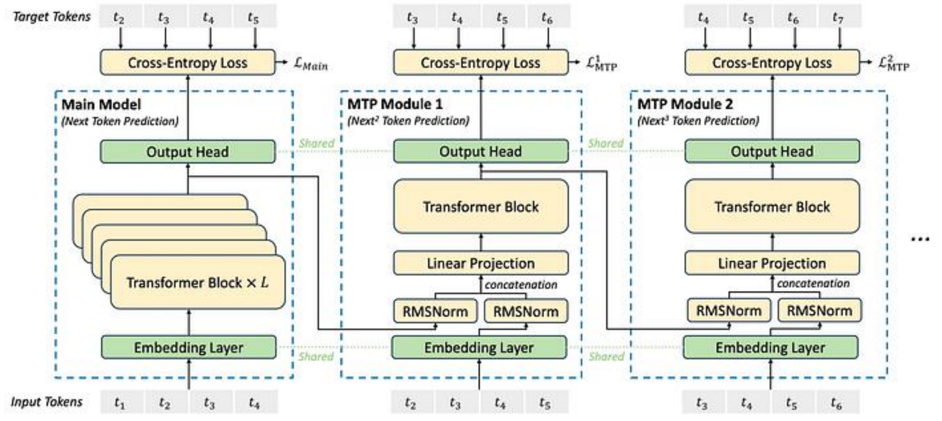
The first phase begins with SFT, which provides a foundation to overcome cold-start issues in RL-only training. Instead of using traditional human-labeled data, the model generated few-shot demonstrations, which were then refined by human reviewers. Thousands of high-quality reasoning examples were collected for this stage. Once the model was fine-tuned on this cold-start data, it underwent large-scale RL training, following the same GRPO framework used in R1-Zero. A language consistency reward was introduced in this phase to reduce language mixing by penalizing responses with inconsistent linguistic patterns.

Following the reinforcement learning phase, DeepSeek-R1 underwent a second SFT stage with a more diverse dataset. Unlike the initial fine-tuning, which primarily focused on reasoning, this stage improved general writing, role-playing tasks, and creative storytelling. **Instead of relying on human-annotated responses, the model used a rejection sampling approach where responses were evaluated using a model-as-a-judge system, ensuring that only high-scoring responses were retained for further training**. The final stage of training involved another round of reinforcement learning, this time refining helpfulness, harmlessness, and reasoning quality across various types of prompts. This phase incorporated data from multiple sources to ensure the model could adapt to different conversational styles and maintain consistency across its responses.

DeepSeek-V3: The architectural foundation of DeepSeek-R1

DeepSeek-R1 leverages the architectural innovations of DeepSeek-V3, which is optimized for scalability and efficiency. **The Multi-Head Latent Attention (MLA) mechanism significantly reduces memory usage by compressing Key-Value (KV) caches, enabling the model to handle 128K-token contexts efficiently**. **The DeepSeekMoE architecture optimizes expert routing with 256 experts per layer, with eight activated per token, preventing expert overloading while improving specialization. Additionally, the Multi-Token Prediction (MTP) mechanism allows the model to predict multiple future tokens simultaneously, accelerating training by 1.8× and improving the overall efficiency.**

Figure 24: Multi-token prediction



SOURCE: DEEPSEEK PAPER,INCRED RESEARCH

These optimizations make DeepSeek-V3 an ideal foundation for reinforcement learning-based training, allowing DeepSeek-R1 to outperform traditional RLHF-based models with lower computational costs.

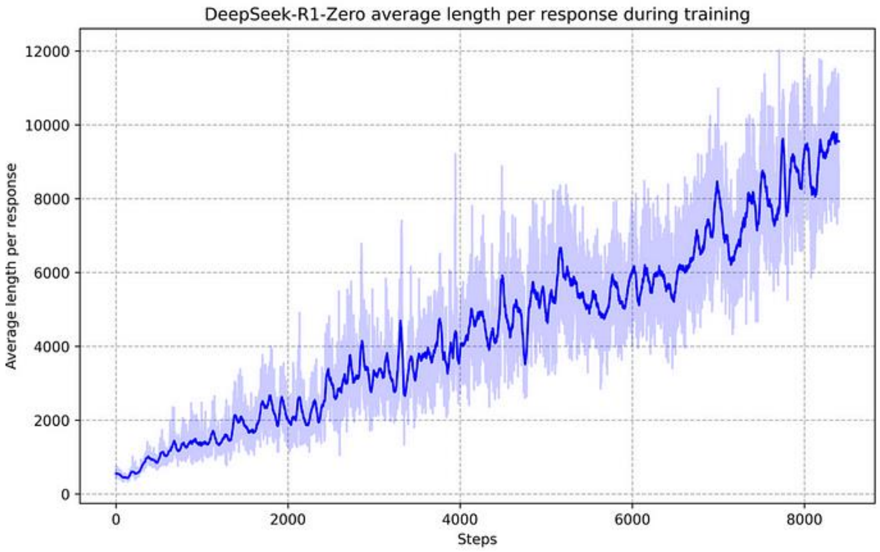
DeepSeek-V3 is also supported by extensive training infrastructure optimizations. The model is trained using 2048 NVIDIA H800 graphics processing units or GPUs, with FP8 mixed-precision training reducing computational overhead by 50% while maintaining performance. **Advanced parallelism strategies, such as expert parallelism and pipeline parallelism, further enhance training speed and scalability**. These infrastructure-level enhancements contribute to DeepSeek-R1’s ability to deliver cutting-edge reasoning performance at a significantly reduced cost.

Performance and real-world impact of DeepSeek-R1

DeepSeek-R1 has demonstrated state-of-the-art performance across multiple domains. In mathematical reasoning, the model achieves 90.2% on MATH-500, outperforming GPT-4. In coding, it excels in LiveCodeBench, surpassing leading models in competitive programming. Additionally, in general knowledge and reasoning, it matches GPT-4 and Claude-3.5 on MMLU benchmarks. Beyond benchmarks, DeepSeek-R1 has proven to be highly effective in real-world applications, excelling in long-form reasoning, handling complex scientific queries, and generating optimized and structured code for software development.

One of the most impressive aspects of DeepSeek-R1 is its ability to leverage test-time computation scaling. **Unlike conventional models that generate responses in a single pass, DeepSeek-R1 iteratively refines its reasoning steps, effectively "thinking out loud" before providing a final answer**. This capability significantly improves the accuracy of its predictions in complex tasks, making it particularly valuable for applications in scientific research, legal analysis, and high-level decision-making.

Figure 25: The average response length of DeepSeek-R1-Zero on the training set during the RL process



SOURCE: DEEPSEEK PAPER,INCRED RESEARCH

The Future of AI Training: Lessons from DeepSeek-R1 ›

The significance of DeepSeek-R1 extends beyond its performance metrics. **It represents a paradigm shift in LLM training by demonstrating that Reinforcement Learning alone can produce competitive models**. Its success challenges existing assumptions about the necessity of large-scale supervised datasets, suggesting that high-quality reasoning can emerge purely through RL-based optimization. Furthermore, DeepSeek-R1 highlights the cost-effectiveness of AI training, achieving GPT-4-level performance at a fraction of the cost. This raises important questions about the future of AI reasoning, particularly whether test-time computation scaling could enable models to surpass human reasoning in specialized domains.

DeepSeek-R1's success also underscores the potential for more efficient and scalable AI training methodologies. By minimizing reliance on human annotation and maximizing the model's ability to learn autonomously, DeepSeek-AI has paved the way for future research into self-improving AI systems. This breakthrough is expected to reshape LLM research, pushing the field toward more efficient, resource-conscious, and scalable AI systems. As the AI community continues to refine reinforcement learning techniques, **DeepSeek-R1 stands as a testament to the power of algorithmic innovation in driving the next generation of AI advancements.**

Comparing ChatGPT and DeepSeek

When comparing ChatGPT and DeepSeek, one of the most immediate differences lies in their pricing models. ChatGPT offers a free tier based on GPT-3.5, but it comes with limitations such as restricted speed during peak hours, daily usage caps, and no access to GPT-4’s advanced capabilities. While this is sufficient for casual users, professionals and developers often find it inadequate for high-quality, consistent output. In contrast, DeepSeek provides unrestricted access to its core model entirely for free, with no feature limitations or payment requirements. This makes DeepSeek particularly appealing to students, small businesses, and researchers who need scalable AI support without financial barriers.

DeepSeek costs 80-90% less than ChatGPT ›

Beyond free-tier access, API pricing plays a crucial role in determining cost efficiency for businesses and developers choosing between ChatGPT and DeepSeek. **OpenAI’s ChatGPT API, particularly for GPT-4, charges US$0.03 per 1,000 input tokens and US$0.06 per 1,000 output tokens. This means generating a single 500-word response costs approximately 6.75 cents, amounting to US$675 per month for 10,000 such responses**. In contrast, DeepSeek offers a significantly more cost-effective model. While its free tier provides unrestricted access for smaller-scale use, its **paid API is estimated to be as low as US$0.01 per 1,000 tokens. This results in just 1.5 cents per 500-word response, translating to US$150 per month for the same usage—a nearly 80% cost reduction compared to ChatGPT**. For businesses with high API demands, DeepSeek presents a more budget-friendly alternative without sacrificing accessibility.

Scalability is a key factor when evaluating API costs, especially for businesses handling large volumes of queries. **ChatGPT’s pricing structure scales quickly, with a 10-million-token monthly usage costing around US$900, and a 100-million-token usage soaring to US$9,000**. This makes it a considerable expense for companies relying on high-frequency AI interactions. In contrast, DeepSeek's estimated pricing model offers a significantly lower cost, with **10 million tokens costing just US$200 per month and 100m tokens totalling US$2,000—a nearly 80% reduction in scaling costs**. For enterprises prioritizing affordability while scaling their AI-driven applications, DeepSeek presents a more economical alternative without compromising performance.

For startups and development teams with sustained AI usage, long-term cost considerations become critical. **With ChatGPT’s API pricing, a business processing 100m tokens per month would incur an annual expense of US$108,000. In contrast, DeepSeek’s lower-cost model would bring the yearly total down to just US$24,000—resulting in an US$84,000 savings**. This substantial cost difference makes DeepSeek a far more budget-friendly option for companies looking to scale AI-driven applications without significantly increasing operational expenses.

A key differentiator between DeepSeek and ChatGPT lies in their approach to accessibility—**DeepSeek is open source, while ChatGPT remains proprietary**. DeepSeek’s open-source nature enables businesses to self-host the model, offering complete control over data privacy, eliminating recurring API costs, and allowing custom optimizations tailored to specific industries. In contrast, ChatGPT’s proprietary model locks users into OpenAI’s infrastructure and pricing, restricting flexibility and potentially leading to higher long-term costs. For organizations prioritizing customization and cost efficiency, DeepSeek provides a compelling alternative.

DeepSeek at par or better than most advanced models ➤

Knowledge

DeepSeek-V3 sets a new standard among open-source models on key educational benchmarks, scoring 88.5 on MMLU, 75.9 on MMLU-Pro, and 59.1 on GPQA—outperforming all other open models. Its results are comparable to top closed-source models like GPT-4o and Claude-Sonnet-3.5, effectively narrowing the gap between open and closed AI systems.

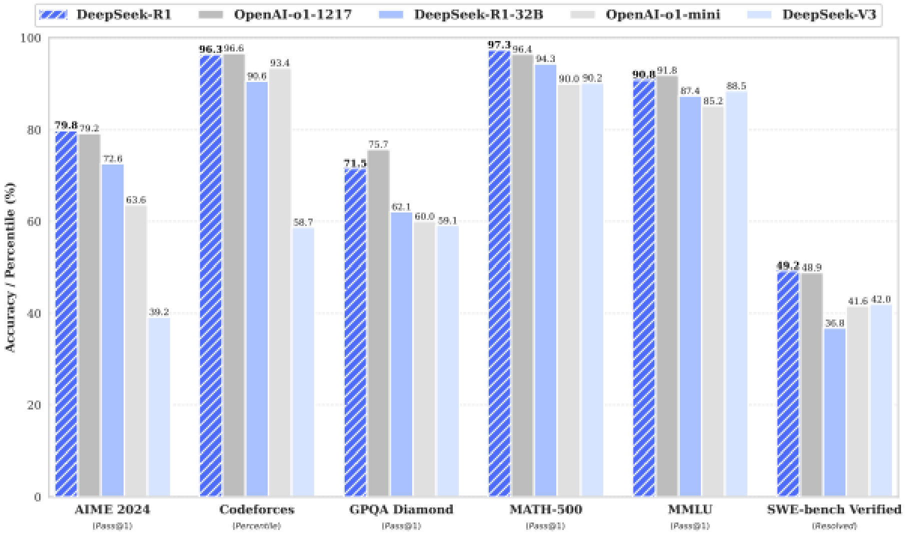
In factuality benchmarks, DeepSeek-V3 leads among open-source models in both SimpleQA and Chinese SimpleQA. While it lags behind GPT-4o and Claude-Sonnet-3.5 in English factual knowledge (SimpleQA), it surpasses them in Chinese factual knowledge, underscoring its strength in this area.

Code, math, and reasoning

DeepSeek-V3 delivers state-of-the-art performance in mathematical reasoning among all non-long-CoT open and closed-source models. It even outperforms o1-preview on certain benchmarks, such as MATH-500, highlighting its advanced problem-solving capabilities.

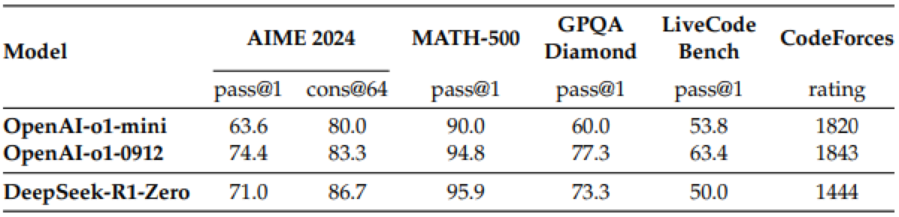
In coding, DeepSeek-V3 dominates competition benchmarks like LiveCodeBench, making it the top-performing model in this domain. For engineering tasks, it slightly trails Claude-Sonnet-3.5 but maintains a significant lead over all other models, demonstrating its broad technical expertise.

Figure 26: Benchmark performance of DeepSeek-R1



SOURCE:INCRED RESEARCH

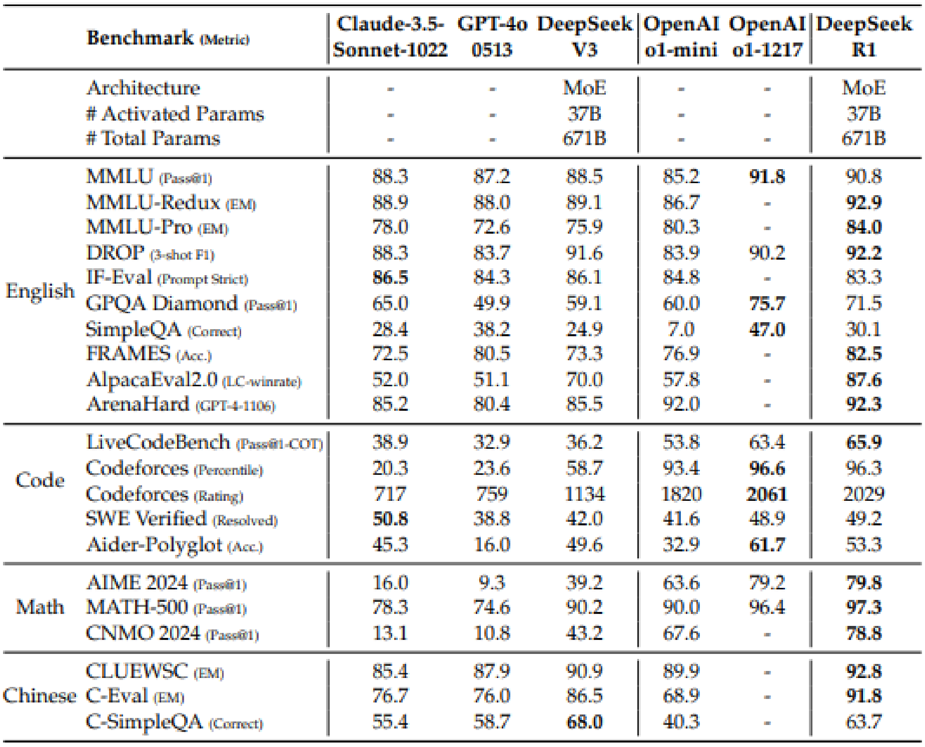
Figure 27: Comparison of DeepSeek-R1-Zero and OpenAI o1 models on reasoning-related benchmarks



Model	AIME 2024		MATH-500	GPQA Diamond	LiveCode Bench	CodeForces
	pass@1	cons@64	pass@1	pass@1	pass@1	rating
OpenAI-o1-mini	63.6	80.0	90.0	60.0	53.8	1820
OpenAI-o1-0912	74.4	83.3	94.8	77.3	63.4	1843
DeepSeek-R1-Zero	71.0	86.7	95.9	73.3	50.0	1444

SOURCE:INCRED RESEARCH

Figure 28: Comparison between DeepSeek-R1 and other representative models



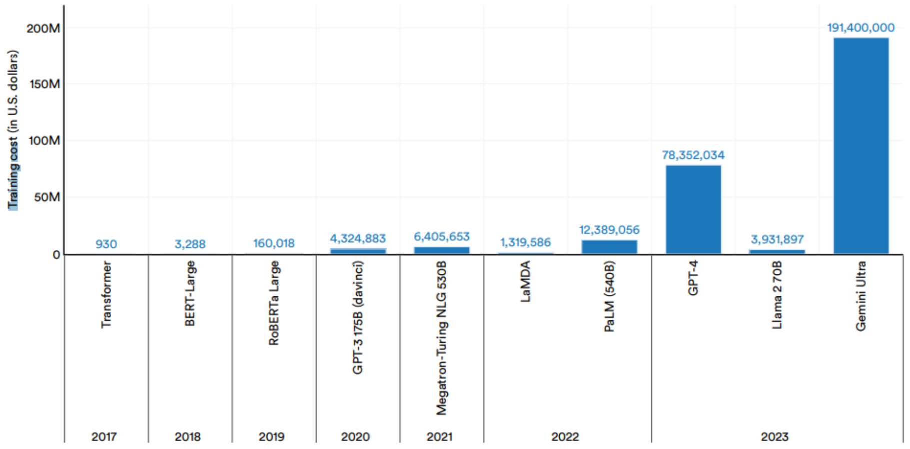
	Benchmark (Metric)	Claude-3.5-Sonnet-1022	GPT-4o 0513	DeepSeek V3	OpenAI o1-mini	OpenAI o1-1217	DeepSeek R1
	Architecture	-	-	MoE	-	-	MoE
	# Activated Params	-	-	37B	-	-	37B
	# Total Params	-	-	671B	-	-	671B
English	MMLU (Pass@1)	88.3	87.2	88.5	85.2	**91.8**	90.8
	MMLU-Redux (EM)	88.9	88.0	89.1	86.7	-	**92.9**
	MMLU-Pro (EM)	78.0	72.6	75.9	80.3	-	**84.0**
	DROP (3-shot F1)	88.3	83.7	91.6	83.9	90.2	**92.2**
	IF-Eval (Prompt Strict)	**86.5**	84.3	86.1	84.8	-	83.3
	GPQA Diamond (Pass@1)	65.0	49.9	59.1	60.0	**75.7**	71.5
	SimpleQA (Correct)	28.4	38.2	24.9	7.0	**47.0**	30.1
	FRAMES (Acc.)	72.5	80.5	73.3	76.9	-	**82.5**
	AlpacaEval2.0 (LC-winrate)	52.0	51.1	70.0	57.8	-	**87.6**
	ArenaHard (GPT-4-1106)	85.2	80.4	85.5	92.0	-	**92.3**
Code	LiveCodeBench (Pass@1-COT)	38.9	32.9	36.2	53.8	63.4	**65.9**
	Codeforces (Percentile)	20.3	23.6	58.7	93.4	**96.6**	96.3
	Codeforces (Rating)	717	759	1134	1820	**2061**	2029
	SWE Verified (Resolved)	**50.8**	38.8	42.0	41.6	48.9	49.2
	Aider-Polyglot (Acc.)	45.3	16.0	49.6	32.9	**61.7**	53.3
Math	AIME 2024 (Pass@1)	16.0	9.3	39.2	63.6	79.2	**79.8**
	MATH-500 (Pass@1)	78.3	74.6	90.2	90.0	96.4	**97.3**
	CNMO 2024 (Pass@1)	13.1	10.8	43.2	67.6	-	**78.8**
Chinese	CLUEWSC (EM)	85.4	87.9	90.9	89.9	-	**92.8**
	C-Eval (EM)	76.7	76.0	86.5	68.9	-	**91.8**
	C-SimpleQA (Correct)	55.4	58.7	**68.0**	40.3	-	63.7

SOURCE: INCRED RESEARCH

DeepSeek's power consumption comparison and analysis ➤

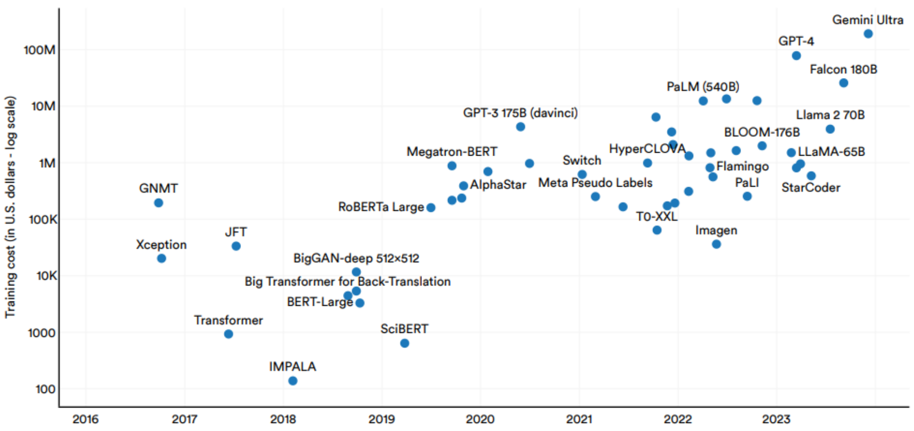
DeepSeek-R1 challenges the conventional belief that larger AI models require exponentially higher energy and financial investments. With a **training cost of just US$5.5m**, it consumes **93% less power than GPT-4 (US$78.3m) and 97% less than Gemini Ultra (US$191.4m)** while delivering **comparable performance**. The efficiency gain extends beyond training—DeepSeek-R1's inference cost is US**$0.55/m input tokens**, an astonishing **97% lower than OpenAI's US$15/m tokens**, significantly reducing run-time energy consumption. In contrast, models like GPT-3 (175B) consumed **1,287 MWh**, and LLaMA 2 (70B) required **430 MWh**, contributing to substantial carbon emissions. DeepSeek-R1's cost-to-performance ratio highlights a major breakthrough in AI sustainability, proving that cutting-edge models can scale without unsustainable energy demand.

Figure 29: Estimated training costs of select AI models



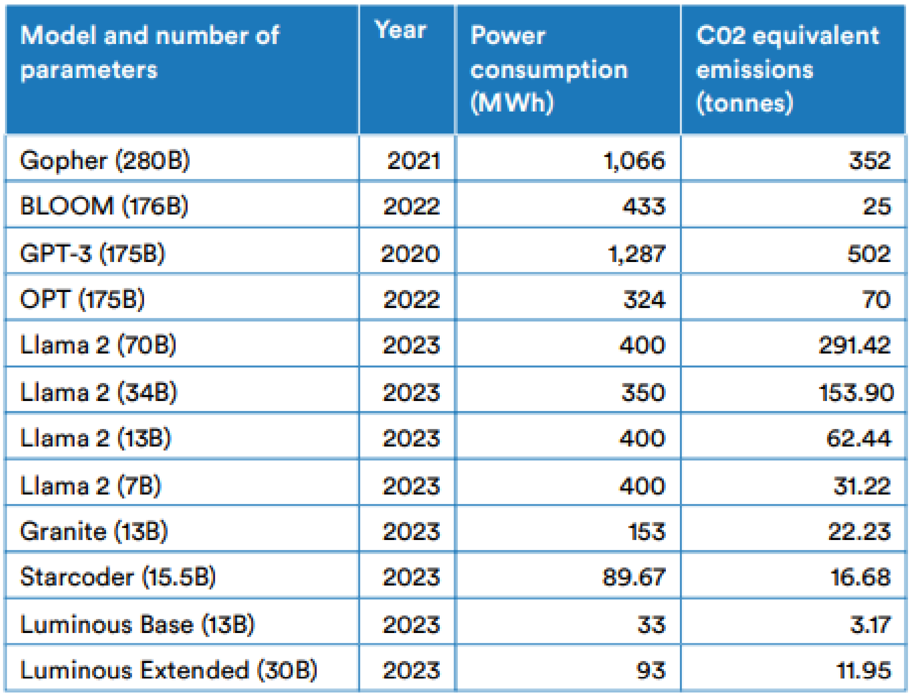
SOURCE: EPOCH, 2023,STANFORD 2024 AI INDEX REPORT,INCRED RESEARCH

Figure 30: Estimated training costs of select AI models



SOURCE: EPOCH, 2023, STANFORD 2024 AI INDEX REPORT,INCRED RESEARCH

Figure 31: Power consumption/environmental impact of select models



Model and number of parameters	Year	Power consumption (MWh)	C02 equivalent emissions (tonnes)
Gopher (280B)	2021	1,066	352
BLOOM (176B)	2022	433	25
GPT-3 (175B)	2020	1,287	502
OPT (175B)	2022	324	70
Llama 2 (70B)	2023	400	291.42
Llama 2 (34B)	2023	350	153.90
Llama 2 (13B)	2023	400	62.44
Llama 2 (7B)	2023	400	31.22
Granite (13B)	2023	153	22.23
Starcoder (15.5B)	2023	89.67	16.68
Luminous Base (13B)	2023	33	3.17
Luminous Extended (30B)	2023	93	11.95

SOURCE: EPOCH, 2023, STANFORD 2024 AI INDEX REPORT, INCRED RESEARCH

DISCLAIMER

This report (including the views and opinions expressed therein, and the information comprised therein) has been prepared by Incred Research Services Private Ltd. (formerly known as Earnest Innovation Partners Private Limited) (hereinafter referred to as "IRSPL"). IRSPL is registered with SEBI as a Research Analyst vide Registration No. INH000011024. Pursuant to a trademark agreement, IRSPL has adopted "Incred Equities" as its trademark for use in this report.

The term "IRSPL" shall, unless the context otherwise requires, mean IRSPL and its affiliates, subsidiaries and related companies. This report is not directed or intended for distribution to or use by any person or entity resident in a state, country or any jurisdiction, where such distribution, publication, availability or use would be contrary to law, regulation or which would subject IRSPL and its affiliates/group companies to registration or licensing requirements within such jurisdictions.

This report is being supplied to you strictly on the basis that it will remain confidential. No part of this report may be (i) copied, photocopied, duplicated, stored or reproduced in any form by any means; or (ii) redistributed or passed on, directly or indirectly, to any other person in whole or in part, for any purpose without the prior written consent of IRSPL.

The information contained in this report is prepared from data believed to be correct and reliable at the time of issue of this report.

IRSPL is not required to issue regular reports on the subject matter of this report at any frequency and it may cease to do so or change the periodicity of reports at any time. IRSPL is not under any obligation to update this report in the event of a material change to the information contained in this report. IRSPL has not any and will not accept any, obligation to (i) check or ensure that the contents of this report remain current, reliable or relevant; (ii) ensure that the content of this report constitutes all the information a prospective investor may require; (iii) ensure the adequacy, accuracy, completeness, reliability or fairness of any views, opinions and information, and accordingly, IRSPL and its affiliates/group companies (and their respective directors, associates, connected persons and/or employees) shall not be liable in any manner whatsoever for any consequences (including but not limited to any direct, indirect or consequential losses, loss of profits and damages) of any reliance thereon or usage thereof.

Unless otherwise specified, this report is based upon reasonable sources. Such sources will, unless otherwise specified, for market data, be market data and prices available from the main stock exchange or market where the relevant security is listed, or, where appropriate, any other market. Information on the accounts and business of company(ies) will generally be based on published statements of the company(ies), information disseminated by regulatory information services, other publicly available information and information resulting from our research. While every effort is made to ensure that statements of facts made in this report are accurate, all estimates, projections, forecasts, expressions of opinion and other subjective judgments contained in this report are based on assumptions considered to be reasonable as of the date of the document in which they are contained and must not be construed as a representation that the matters referred to therein will occur. Past performance is not a reliable indicator of future performance. The value of investments may go down as well as up and those investing may, depending on the investments in question, lose more than the initial investment. No report shall constitute an offer or an invitation by or on behalf of IRSPL and its affiliates/group companies to any person to buy or sell any investments.

The opinions expressed are based on information which is believed to be accurate and complete and obtained through reliable public or other non-confidential sources at the time made (information barriers and other arrangements may be established, where necessary, to prevent conflicts of interests arising. However, the analyst(s) may receive compensation that is based on his/their coverage of company(ies) in the performance of his/their duties or the performance of his/their recommendations. In reviewing this report, an investor should be aware that any or all of the foregoing, among other things, may give rise to real or potential conflicts of interest. Additional information is, subject to the duties of confidentiality, available on request.

The report is not a "prospectus" as defined under Indian Law, including the Companies Act, 2013, and is not, and shall not be, approved by, or filed or registered with, any Indian regulator, including any Registrar of Companies in India, SEBI, any Indian stock exchange, or the Reserve Bank of India. No offer, or invitation to offer, or solicitation of subscription with respect to any such securities listed or proposed to be listed in India is being made, or intended to be made, to the public, or to any member or section of the public in India, through or pursuant to this report.

The research analysts, strategists or economists principally responsible for the preparation of this research report are segregated from the other activities of IRSPL. Information barriers and other arrangements have been established, as required, to prevent any conflicts of interests.

The research analysts, strategists or economists principally responsible for the preparation of this research report are segregated from the other activities of IRSPL. Information barriers and other arrangements have been established, as required, to prevent any conflicts of interests.

IRSPL may have issued other reports (based on technical analysis, event specific, short-term views, etc.) that are inconsistent with and reach a different conclusion from the information presented in this report.

Holding of Analysts/Relatives of Analysts, IRSPL and Associates of IRSPL in the covered securities, as on the date of publishing of this report

Research Analyst or his/her relative(s) or InCred Research Services Private Limited or our associate may have any financial interest in the subject company.

Research Analyst or his/her relatives or InCred Research Services Limited or our associates may have actual or beneficial ownership of 1% or more securities of the subject company(ies) at the end of the month immediately preceding the date of publication of the Research Report.

Research Analyst or his/her relative or InCred Research Services Private Limited or our associate entities may have any other material conflict of interest at the time of publication of the Research Report.

In the past 12 months, IRSPL or any of its associates may have:

a) Received any compensation/other benefits from the subject company,
b) Managed or co-managed public offering of securities for the subject company,
c) Received compensation for investment banking or merchant banking or brokerage services from the subject company,
d) Received compensation for products or services other than investment banking or merchant banking or brokerage services from the subject company

We or our associates may have received compensation or other benefits from the subject company(ies) or third party in connection with the research report.

Research Analyst may have served as director, officer, or employee in the subject company.

We or our research analyst may engage in market-making activity of the subject company.

Analyst declaration

- The analyst responsible for the production of this report hereby certifies that the views expressed herein accurately and exclusively reflect his or her personal views and opinions about any and all of the issuers or securities analysed in this report and were prepared independently and autonomously in an unbiased manner.
- No part of the compensation of the analyst(s) was, is, or will be directly or indirectly related to the inclusion of specific recommendations(s) or view(s) in this report or based on any specific investment banking transaction.
- The analyst(s) has(have) not had any serious disciplinary action taken against him/her(them).
- The analyst, strategist, or economist does not have any material conflict of interest at the time of publication of this report.
- The analyst(s) has(have) received compensation based upon various factors, including quality, accuracy and value of research, overall firm performance, client feedback and competitive factors.

IRSPL and/or its affiliates and/or its Directors/employees may own or have positions in securities of the company(ies) covered in this report or any securities related thereto and may from time to time add to or dispose of, or may be materially interested in, any such securities.

IRSPL and/or its affiliates and/or its Directors/employees may do and seek to do business with the company(ies) covered in this research report and may from time to time (a) buy/sell the securities covered in this report, from time to time and/or (b) act as market maker or have assumed an underwriting commitment in securities of such company(ies), and/or (c) may sell them to or buy them from customers on a principal basis and/or (d) may also perform or seek to perform significant investment banking, advisory, underwriting or placement services for or relating to such company(ies) and/or (e) solicit such investment, advisory or other services from any entity mentioned in this report and/or (f) act as a lender/borrower to such company and may earn brokerage or other compensation. However, Analysts are forbidden to acquire, on their own account or hold securities (physical or uncertificated, including derivatives) of companies in respect of which they are compiling and producing financial recommendations or in the result of which they play a key part.
Registration granted by SEBI, membership of a SEBI recognized supervisory body (if any) and certification from NISM in no way guarantee performance of the intermediary or provide any assurance of returns to investors.

InCred Research Services Private Limited
Research Analyst SEBI Registration Number: INH000011024
Registered Office: Unit No 1203, 12th Floor, B Wing, The Capital, C-70, G Block, BKC, Bandra (E), Mumbai – 400051
Phone: +91-22-6844-6100
Corporate Office: 05th floor, Laxmi Towers, Plot No. C-25, G Block, Bandra – Kurla Complex, Bandra (East), Mumbai – 400051
Phone: +91-22-4161-1500
Name of the Compliance Officer: Mr. Yogesh Kadam
Email ID: compliance@incredresearch.com, Phone No: +91-22-41611539
For any queries or grievances, you may contact the Grievance Officer.
Name of the Grievance Officer: Mr. Rajarshi Maitra
Phone no. +91-022-41611546
Email ID: rajarshi.maitra@incredresearch.com
CIN: U74999MH2016PTC287535

Recommendation Framework

Stock Ratings Definition:

Add The stock’s total return is expected to exceed 10% over the next 12 months.

Hold The stock’s total return is expected to be between 0% and positive 10% over the next 12 months.

Reduce The stock’s total return is expected to fall below 0% or more over the next 12 months.

The total expected return of a stock is defined as the sum of the: (i) percentage difference between the target price and the current price and (ii) the forward net dividend yields of the stock. Stock price targets have an investment horizon of 12 months.

Sector Ratings Definition:

Overweight An Overweight rating means stocks in the sector have, on a market cap-weighted basis, a positive absolute recommendation.

Neutral A Neutral rating means stocks in the sector have, on a market cap-weighted basis, a neutral absolute recommendation.

Underweight An Underweight rating means stocks in the sector have, on a market cap-weighted basis, a negative absolute recommendation.

Country Ratings Definition:

Overweight An Overweight rating means investors should be positioned with an above-market weight in this country relative to benchmark.

Neutral A Neutral rating means investors should be positioned with a neutral weight in this country relative to benchmark.

Underweight An Underweight rating means investors should be positioned with a below-market weight in this country relative to benchmark.