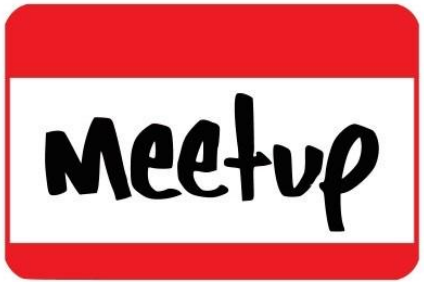
Python Frederick

Overview to Generative AI



June 11th, 2025

7:00pm - 9:00pm

My Top AI Tools

Windsurf
Cursor AI
ChatGPT
Github copilot
Claude
gemini.google.com
perplexity
meta.ai
Manus
bing.com
Microsoft copilot
runnerh.com
notebooklm.google
anythingllm.com
Ollama
lmstudio

https://github.com/gmossy
https://www.linkedin.com/in/gmossy/

Python Frederick Meetup

🤖 Introduction to AI Agents 🤖

GM

Glenn Mossy

AI Agent Architect

37+ years turning sci-fi into production code! 🚀 From nuclear plants to autonomous AI agents

🏗️ **Warp Core AI Autonomy Stack**
Lead architect for autonomous agent framework serving ISR operations with neurosymbolic reasoning

🧠 **ATEM LLM RAG Copilot**
Built comprehensive GenAI knowledge assistant for military technical documentation

🌐 **DISA's First Production SDN**
Delivered software-defined networking system automating critical network management

📞 **Hughes Enterprise VoIP Pioneer**
Launched North America's first satellite-based enterprise voice service

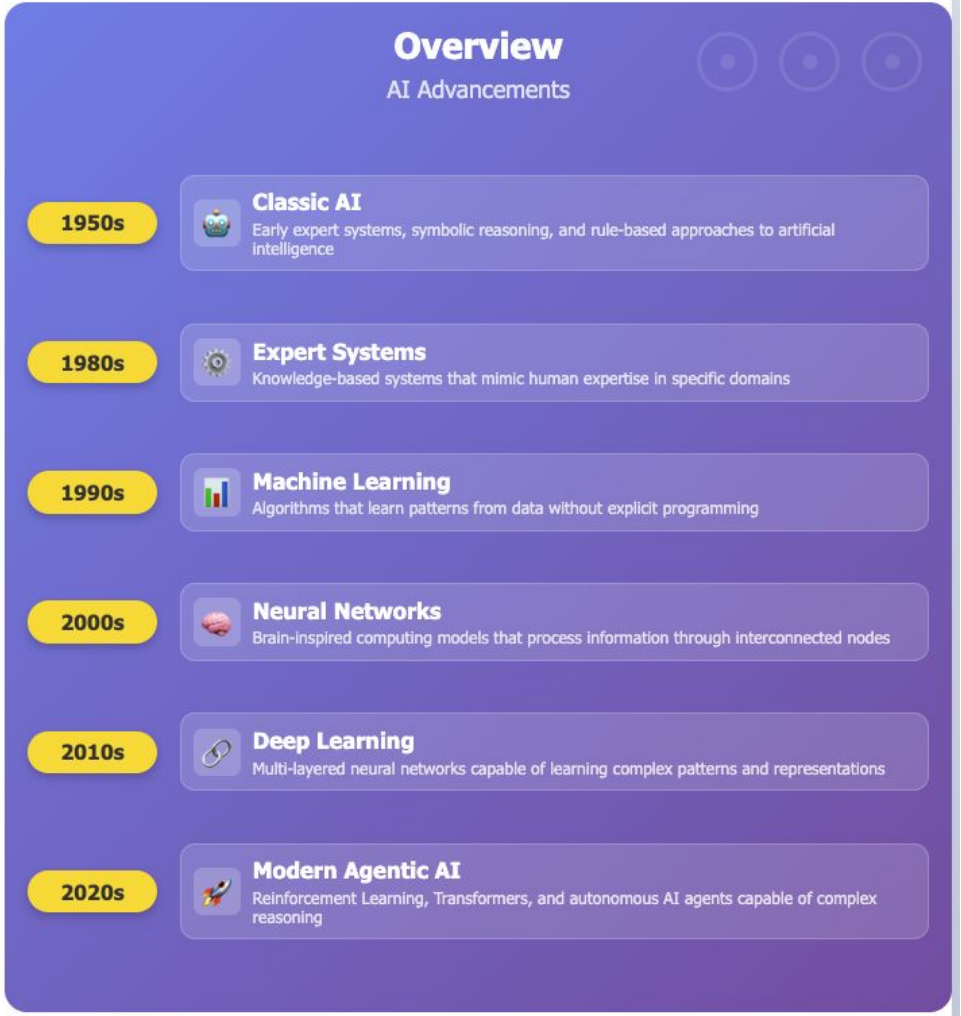
🎯 **Fun Agent Facts**

- Taught 2,000+ adults electronics & robotics at Frederick Community College
- Built security systems for nuclear plants (back when AI was just a dream!)
- Lives in Ijamsville, MD - probably your neighbor! 👋
- Started with Assembly & C, now wielding Python like a wizard 🧙

Python 3.11+ | LangChain | LlamaIndex | AWS Bedrock | PyTorch | Autonomous Agents | RAG Systems | Neurosymbolic AI

GenAI: An Introduction Overview

- Presentation is an Overview to Generative AI
- Includes an Evolution Timeline from AI Foundations to Agentic AI
- Discusses key LLM Architectures and Their Applications
- Covers Challenges and Limitations of Current LLMs
- Explores the Transformer Model and Attention Mechanism



Introduction to Generative AI: The New Electricity

Understanding the Building Blocks of Modern AI Revolution

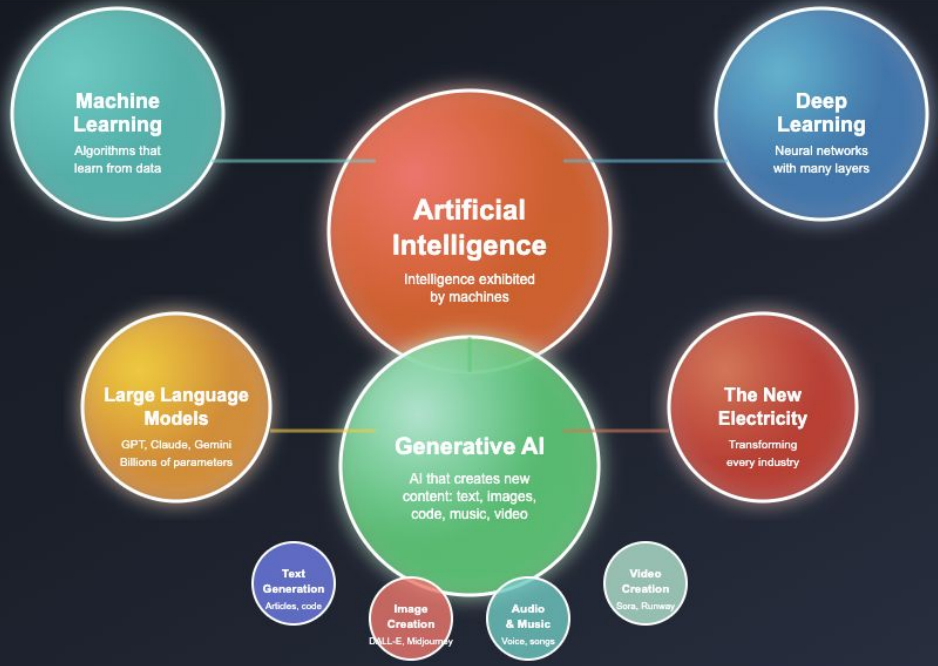
"AI is the new electricity" - Andrew Ng

Just as electricity transformed every industry 100 years ago, AI will now transform every industry

Professor at Stanford University, Co-founder of Coursera, Former Chief Scientist at Baidu

Traditional AI

- Recognizes patterns
- Classifies data
- Makes predictions
- Reactive systems



Generative AI

- Creates new content
- Generates ideas
- Assists creativity
- Proactive partner

Why Generative AI is Revolutionary

- **Creative:** Generates novel content never seen before
- **Multimodal:** Works with text, images, audio, video
- **Transformative:** Changing how we work and create
- **Conversational:** Natural human-like interaction
- **Democratizing:** Making creation accessible to everyone
- **Scalable:** From startups to enterprises

From Generative AI to Agentic AI: Complete Evolution Timeline

Early Foundations → Deep Learning Revolution → Modern GenAI → Autonomous Agentic Systems

🌟 Key Pioneers & Their Contributions

Alan Turing
1912-1954
Machine Intelligence
Turing Test
Computing Foundation

Joseph Weizenbaum
1923-2008
ELIZA Creator
First Chatbot
NLP Pioneer

Geoffrey Hinton
1947-
"Godfather of AI"
Backpropagation
Deep Learning

Yann LeCun
1960-
CNN Pioneer
Computer Vision
Meta AI Chief

Yoshua Bengio
1964-
RNN Expert
Sequence Learning
Turing Award 2018

Ian Goodfellow
1985-
GAN Inventor
Generative AI
Apple AI Director

Dario Amodei
1983-
Anthropic CEO
Claude Creator
AI Safety Focus

🚀 Evolution of AI Capabilities

Early Foundations (1940s-1980s)

Key Concepts:
- Information Theory • Pattern Recognition
- Rule-based Systems • Early Neural Networks
- Symbolic AI • First Chatbots

Deep Learning Era (1986-2010s)

Breakthroughs:
- Backpropagation • CNNs for Vision
- GPU Acceleration • Big Data Training
- ImageNet Competition • Deep Belief Networks

Modern GenAI (2017-2023)

Innovations:
- Transformer Architecture • Large Language Models
- Generative AI • Foundation Models
- GPT Series • Mainstream Adoption

Agentic AI (2024-Present)

Features:
- Autonomous Action • Tool Integration
- Multi-step Reasoning • Goal-oriented Behavior
- Enterprise Automation • MCP Protocol

🎯 Today's Reality: Agentic AI in Action

- **Claude Computer Use:** AI that can see screens and take actions
- **Enterprise Integration:** AI agents in business workflows
- **Autonomous Decision Making:** AI that plans and executes
- **Multi-Agent Systems:** Teams of AI agents working together
- **MCP Protocol:** Standardized tool integration for agents
- **Tool-Using Agents:** AI that interacts with software and APIs

Future of Agentic AI & Humanoid Robotics: 2025-2045 Roadmap

From Today's Basic Robots to Tomorrow's Autonomous Digital Workers & Humanoid Companions

2025 TODAY — Near-Term (2025-2027)

Specialized Agents
- Claude Computer Use
- Basic Humanoids
- Tesla Optimus Gen 2
- MCP Protocol
- Enterprise Pilots

2027 — Mid-Term (2028-2032)

Multi-Agent Era
- Coordinated AI Teams
- Factory Humanoids
- 10,000+ Optimus Units
- Narrow AI Specialists
- Enterprise Adoption

2030

AGI Emergence
- Human-Level AI
- General Purpose Bots
- Home Assistants
- Mass Production
- $20K Price Point

2035 — Long-Term (2033-2040)

Autonomous Society
- Fully Autonomous AI
- Human-Robot Teams
- Emotional Intelligence
- 100M+ Humanoids
- Seamless Integration

2045+ — Future Vision (2040+)

ASI Era
- Superintelligence
- Digital Consciousness
- Bio-Tech Integration
- Post-Human Society
- Singularity

🏢 Industry Leaders & Their Roadmaps

Microsoft
2025 Focus:
- Copilot Agent Ecosystem
- Azure AI Agent Service
- Enterprise Integration
- Multi-Agent Orchestration

OpenAI
AGI by 2027:
- GPT-5 & Beyond
- Operator Agent
- Reasoning Models
- Human-Level AI

Google
Gemini 2.0+:
- Project Mariner
- Agentspace Platform
- MCP Support
- AGI by 2030

Anthropic
Claude Evolution:
- Computer Use Mastery
- MCP Leadership
- Safety-First Approach
- Human-Level by 2027

Meta
Llama Agents:
- Open Source Strategy
- Multimodal Agents
- AR/VR Integration
- Metaverse Agents

🤖 Robotics Evolution: From Basic Bots to Humanoid Companions

Today's Reality
Current Capabilities:
- Tesla Optimus Gen 2: Basic tasks
- Boston Dynamics Atlas: Agility
- Figure-01: Warehouse operations
- Limited autonomy & dexterity
- $50K-$100K+ price range

2027-2030: Mass Production
Expected Progress:
- 10,000+ Optimus units in factories
- Enhanced dexterity (22 DoF hands)
- Basic household tasks
- Price drops to $20K-$30K
- Early home adoption for wealthy

2030-2035: Mainstream
Breakthrough Era:
- Full household integration
- Natural conversation abilities
- Emotional intelligence
- Sub-$20K consumer models
- 100M+ units globally

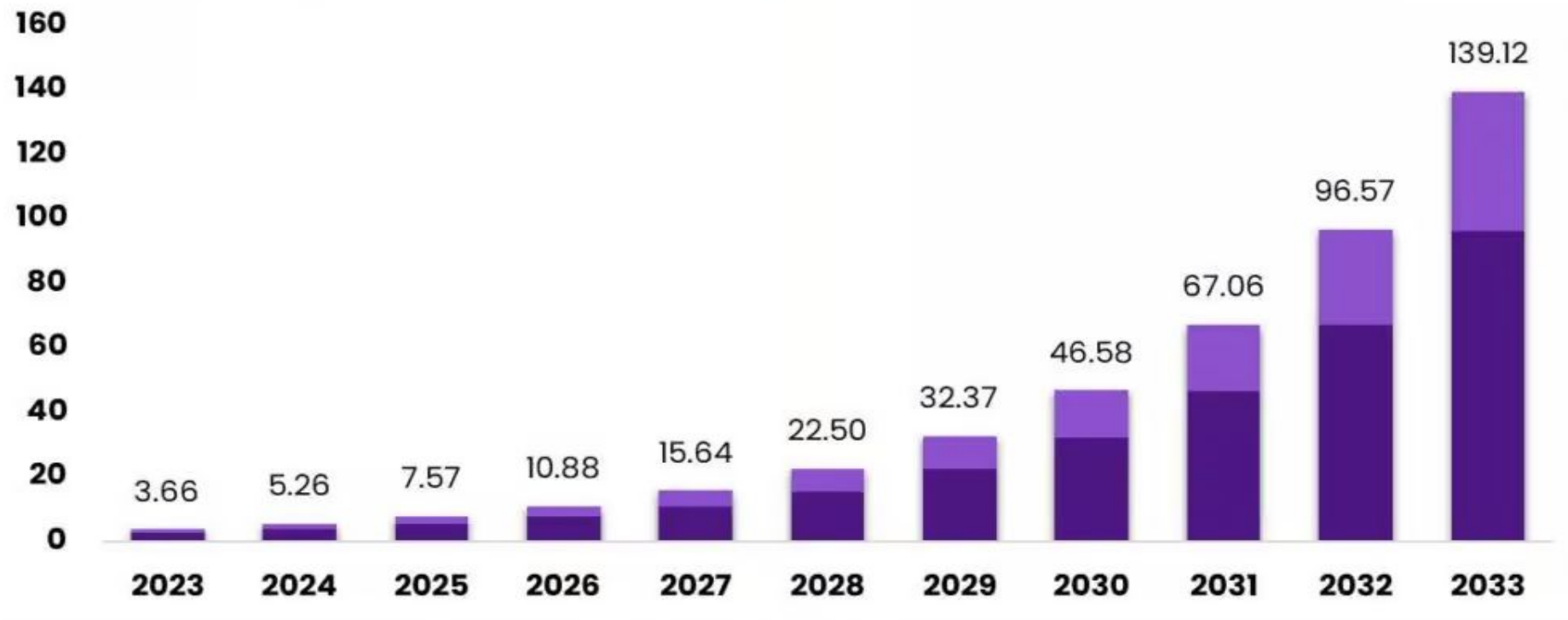
2035+: Companions
Future Vision:
- Human-like appearance & behavior
- Deep emotional bonds
- Creative & artistic abilities
- Integrated into society
- Rights & legal status

🔮 Key Predictions: The Convergence Ahead

- **2027:** AGI achieved by OpenAI/Anthropic - AI matches human cognitive abilities
- **2030:** 100M+ humanoid robots in homes, offices, and factories worldwide
- **2035:** Human-robot teams become the norm in most workplaces
- **2040+:** ASI emergence - AI systems surpass human intelligence in all domains

Global AI Agents Market

Size, by Agent Type, 2024-2033 (USD Billion)



The Market will Grow At the CAGR of: 43.8% **The Forecasted Market Size for 2033 in USD: $139.12B**

market.us
ONE STOP SHOP FOR THE REPORTS

https://market.us/report/ai-agents-market/

FRONTIER AI MODELS + HIGHLIGHTS (MAR/2025)

Microsoft phi
Google Gemma
IBM Granite
Mistral

+ hundreds more...

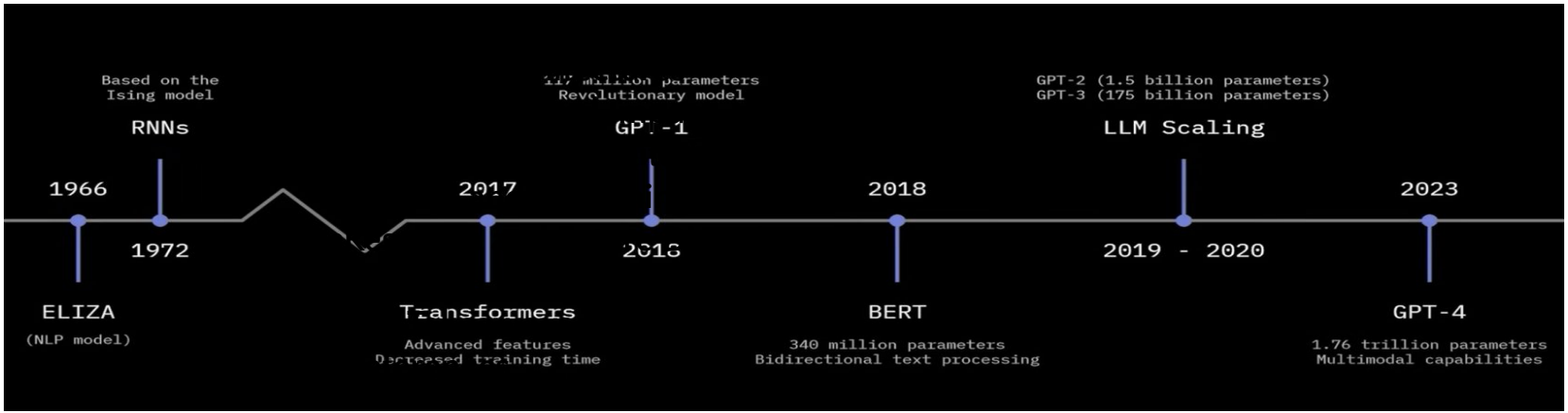
DeepSeek R
Cohere Command-R
AI21 Jamba
Alibaba Qwen/QwQ

+ hundreds more...

Some images by Flaticon.com. Selected highlights only. All 500+ models: https://lifearchitect.ai/models-table/ Alan D. Thompson. 2021–2025.

LifeArchitect.ai/models-table (500+ model highlights)

EVOLUTION TIMELINE OF LANGUAGE



ELIZA: Pioneering Natural Language Processing

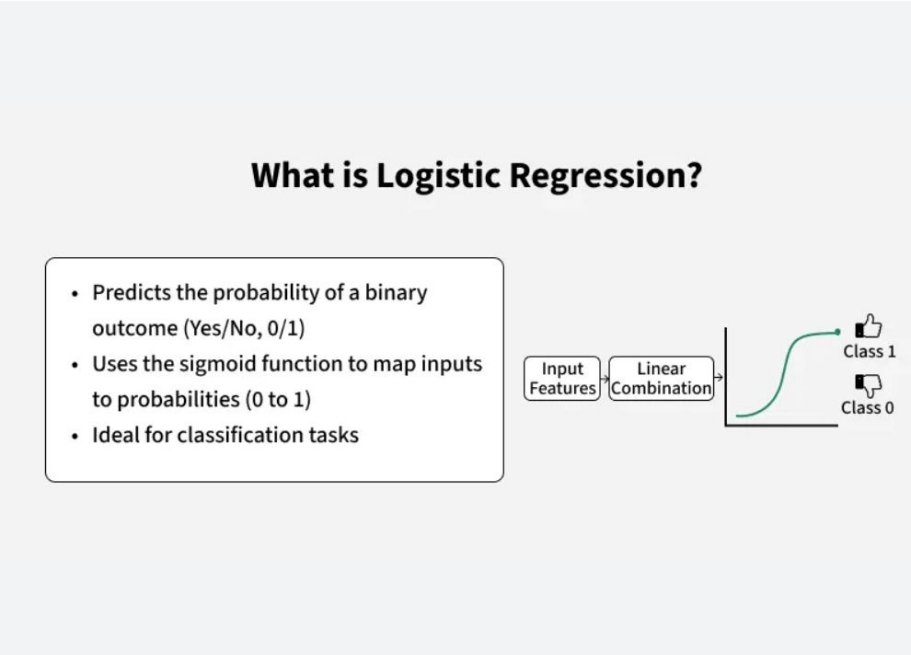
```
                    EEEEEE  LLLLLL  IIII ZZZZZZ    AA   AA

  Eliza is a mock Rogerian psychotherapist.
  The original program was described by Joseph Weizenbaum in 1966.
  This implementation by Norbert Landsteiner 2005.


ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
```

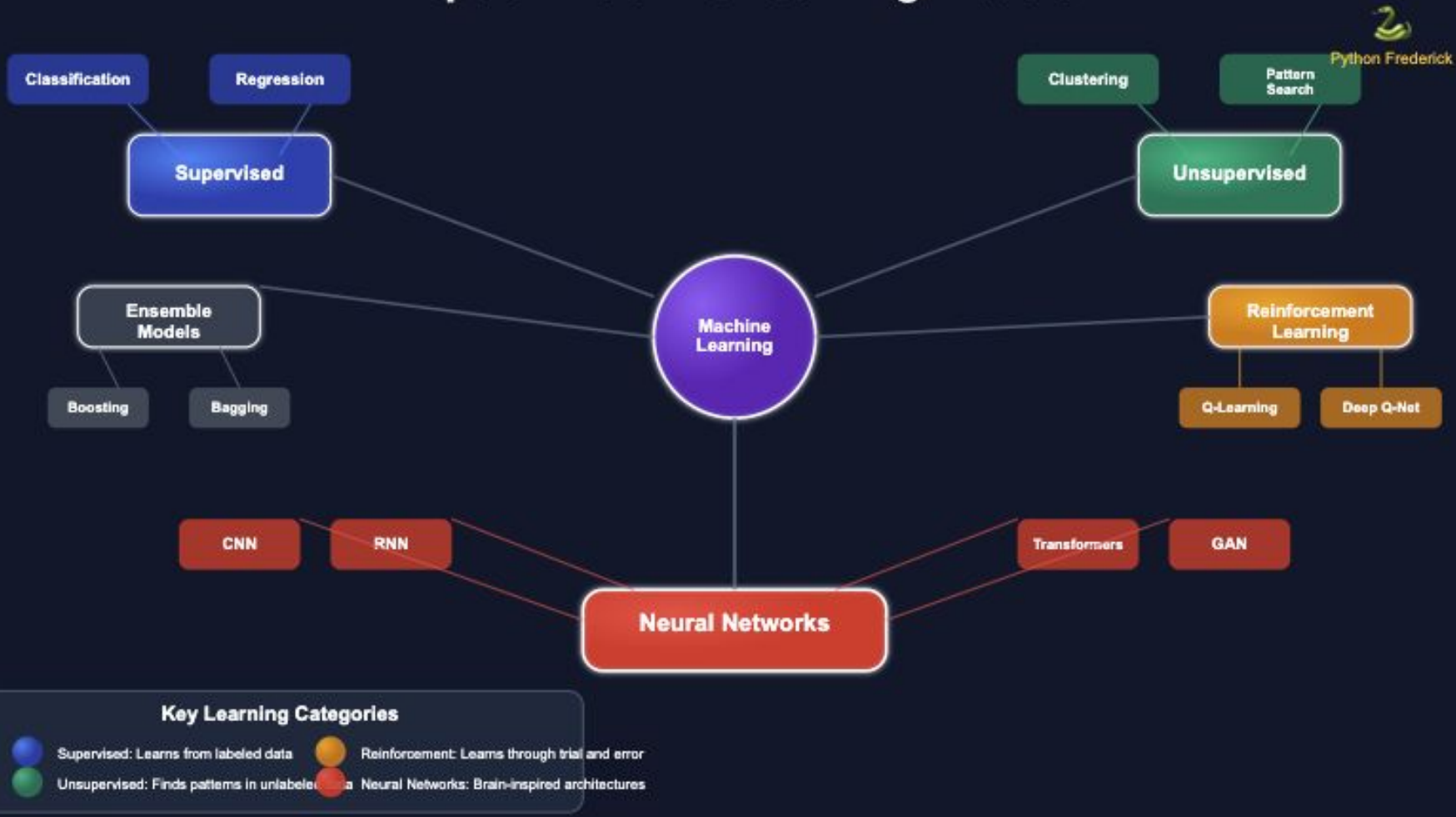
MIT in 1966, Professor Joseph Weizenbaum

Machine Learning



https://www.geeksforgeeks.org/understanding-logistic-regression/

Popular Machine Learning Models

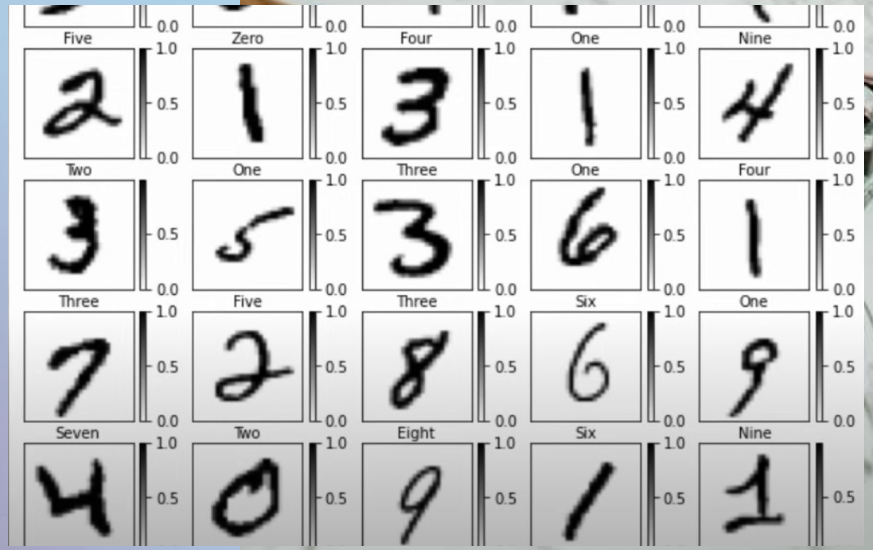


DEEP Learning

CNNs and MINST

The CNN That Started It All: Recognizing Letters with MINST

Convolutional Neural Networks



https://www.geeksforgeeks.org/identifying-handwritten-digits-using-logistic-regression-pytorch/

https://www.geeksforgeeks.org/applying-convolutional-neural-network-on-mnist-dataset/

ATTENTION IS ALL YOU NEED

- https://arxiv.org/pdf/1706.03762

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com
Noam Shazeer* Google Brain noam@google.com
Niki Parmar* Google Research nikip@google.com
Jakob Uszkoreit* Google Research usz@google.com
Llion Jones* Google Research llion@google.com
Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu
Łukasz Kaiser* Google Brain lukaszkaiser@google.com
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

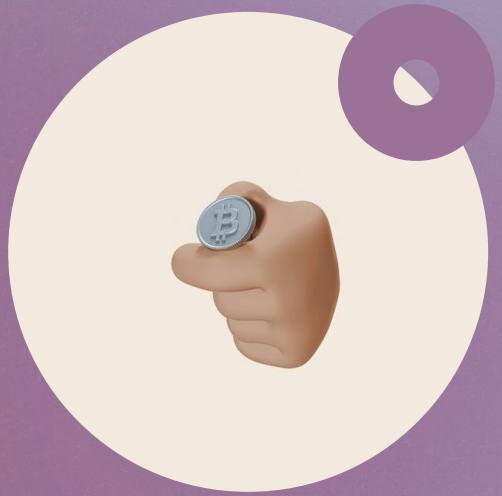
The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

*Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

†Work performed while at Google Brain.

‡Work performed while at Google Research.

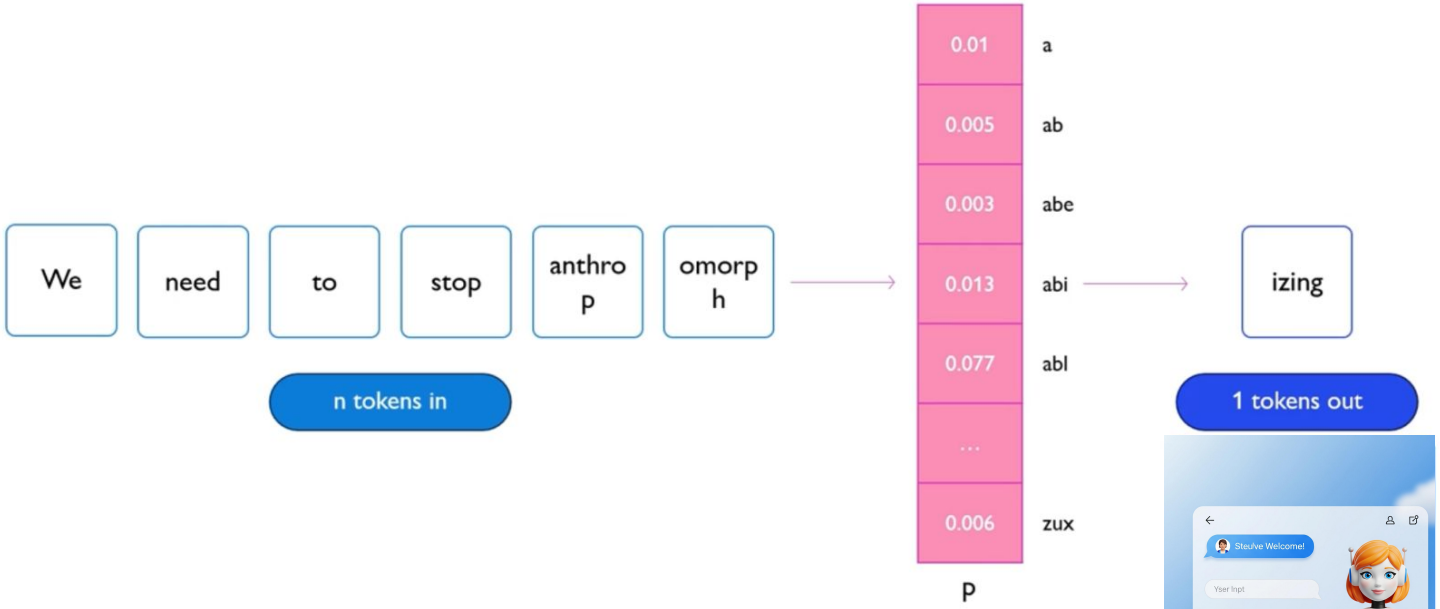
DECODING THE MYSTERY: WHAT EXACTLY ARE LLMS?



Key Facts About Large Language Models (LLMs)

- A Large Language Model (LLM) is an advanced AI that understands and generates human-like text.
- LLMs use transformer neural networks and train on vast text data from various sources.
- They have millions to trillions of parameters to learn complex language patterns.
- Self-attention and deep learning help LLMs create coherent, context-aware text and reason deeply.
- LLMs power chatbots, virtual assistants, content creation, and data analysis tools.

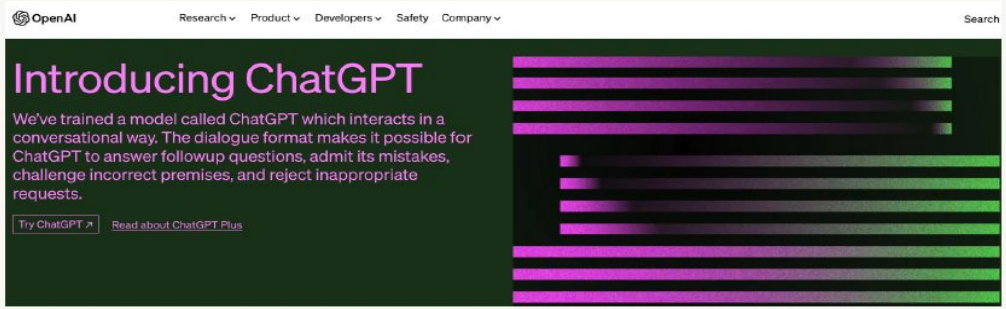
HOW LANGUAGE MODELS GENERATES TEXT



The journey to ChatGPT

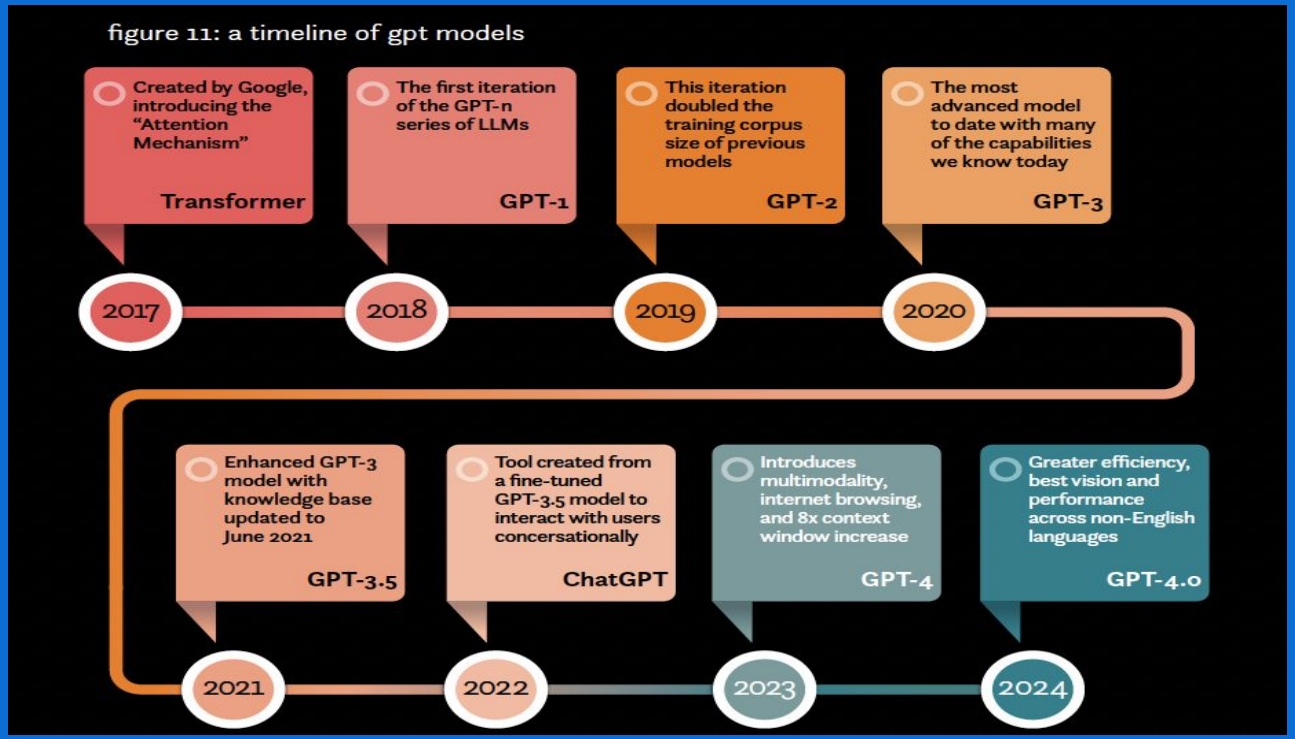
What is GPT?

- ChatGPT was originally built atop a large language model known as GPT-3.5.



- GPT-3.5 is a type of decoder-based transformer model.

TIMELINE OF GPT MODELS



WHAT MAKES LLMS DIFFERENT?

2017
MASSIVE DATA TRAINING

Trained on massive amounts of text data (books, articles, code, the internet).

2018
INNOVATIVE CAPABILITIES

Can generate human-quality text, translate languages, write different kinds of creative content, and answer questions in an informative way.

2019
CREATIVE CONTENT GENERATION

LLMs are used to write stories, poems, code, and marketing copy, showcasing creative abilities.

2020
HUMAN-QUALITY TEXT

WHAT MAKES LLMS DIFFERENT FROM TRADITIONAL COMPUTER PROGRAMS?

2021
EXPANDED LANGUAGE SUPPORT

LLMs begin to support more languages and dialects, increasing accessibility and reach.

2022
IMPROVED INFORMATIVE RESPONSES

LLMs become better at answering complex questions using their vast training data.

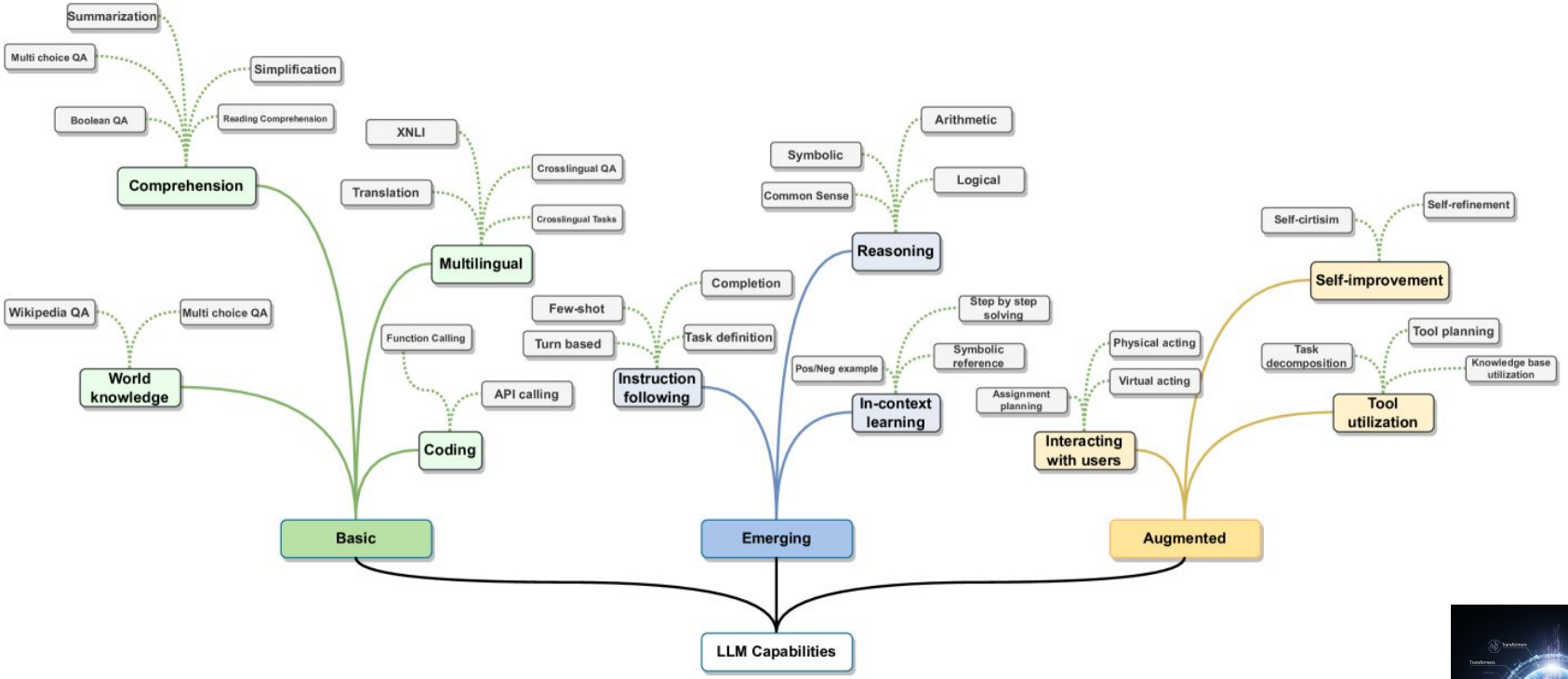
2023
INTEGRATION IN APPLICATIONS

LLMs are integrated into chatbots, virtual assistants, and business tools for diverse uses.

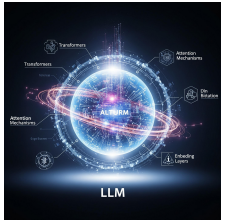
2024
ONGOING ADVANCEMENTS

Continued research leads to even more advanced and capable LLMs, further differentiating them from traditional programs.

LLM CAPABILITIES



https://arxiv.org/html/2402.06196v2



CHALLENGES AND LIMITATIONS OF LLMS

Math, Logic, and Reasoning: Despite being powerful in some areas, LLMs struggle with basic logic, reasoning, and mathematical accuracy compared to humans.

Example: LLMs can make mistakes, like incorrectly counting the number of letters in a string, i.e strawberry

Bias and Safety: LLMs inherit biases from their training data, which is human-created and often flawed. This leads to challenges in ensuring fairness, as well as dealing with harmful or biased outputs. Some companies implement censorship on their models.

Knowledge Cutoff: Historically, LLMs have only been able to draw knowledge from the point they were trained. However, models like ChatGPT and Google Gemini with web-browsing capabilities are beginning to address this limitation.

Hallucinations: LLMs sometimes generate false information with high confidence, a phenomenon known as "hallucination," leading to misinformation .

Also Lack Explainability and Interpretability.

Computational Costs: Training and fine-tuning LLMs require massive computational resources, making it costly to deploy and scale these models.

Environmental Impact is a concern.

Ethical and Legal Concerns: Data Privacy and Security is a big concern.

Misuse of information, and content.

Top 5 Popular LLM APIs (2025)

✅ **1. OpenAI**

- **Models**: GPT-4o, GPT-4-turbo, GPT-3.5
- **Key Features**: Multimodal (text, vision, audio), Assistants API, function calling
- **Docs**: https://platform.openai.com/docs

✅ **2. Anthropic Claude**

- **Models**: Claude 3 Opus, Sonnet, Haiku
- **Key Features**: 200K+ context, safe alignment, structured output
- **Docs**: https://docs.anthropic.com/claude

✅ **3. Google Gemini**

- **Models**: Gemini 1.5 Pro, Gemini 1.0
- **Key Features**: Long-context, vision, tool usage
- **Docs**: https://ai.google.dev/docs

✅ **4. Mistral**

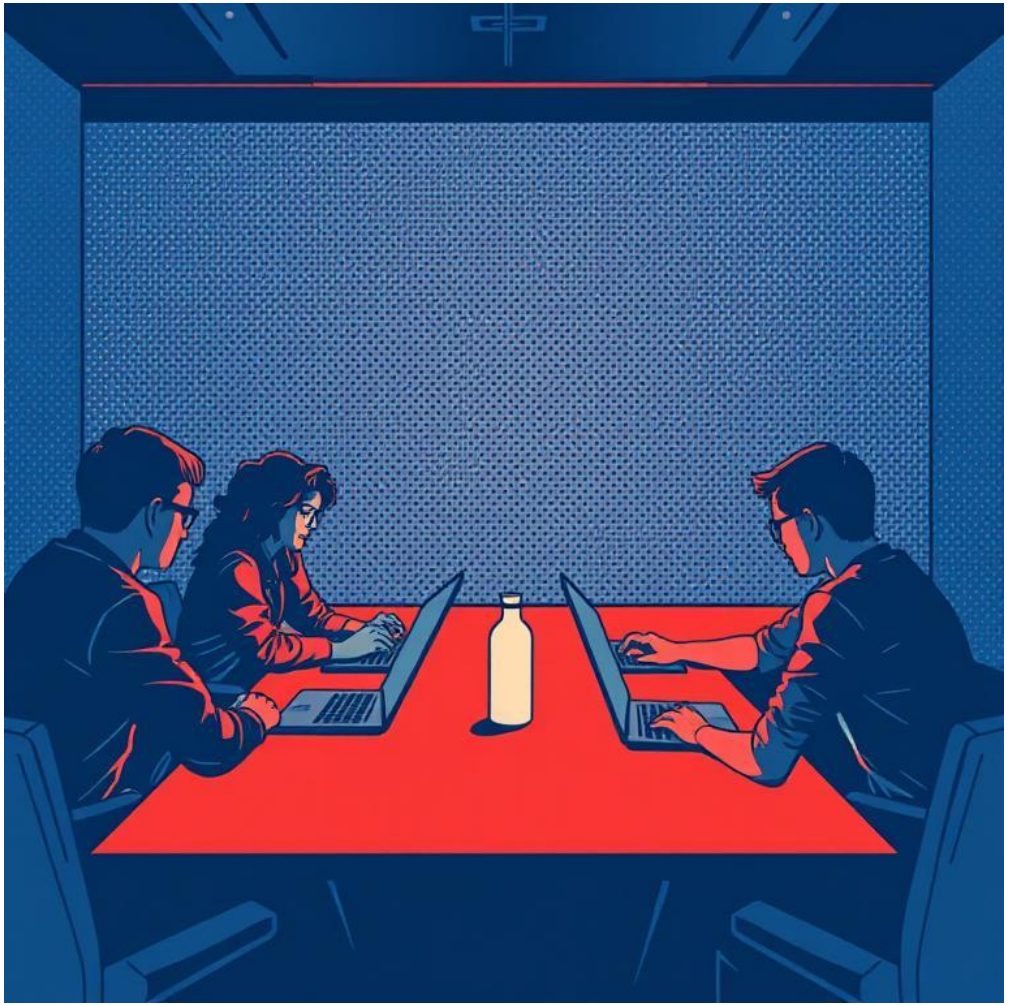
- **Models**: Mistral 7B, Mixtral 8x7B
- **Key Features**: Open-weight + API, function calling, fast inference
- **Docs**: https://docs.mistral.ai

✅ **5. Cohere**

- **Models**: Command R+, Embed v3
- **Key Features**: RAG-native, embeddings, reranking, search

AI: Any Questions?

- What are key differences between AI & GenAI?
- How does transformer architecture enable LLMs?
- What are the ethical concerns of GenAI?
- Can LLMs truly understand human language?



TOKENS

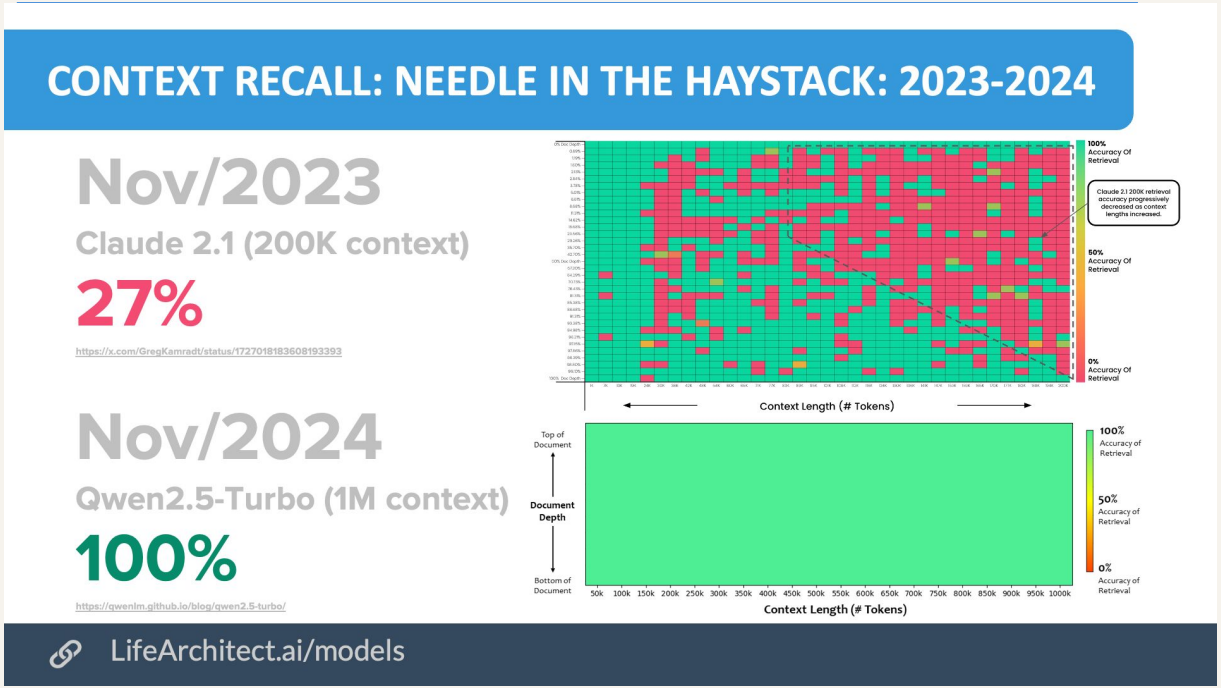
- **What it is:** A token is a way of representing text in a way that an LLM can understand. It's a discrete unit that can be anything from a single character ("a", "!") to a word ("cat", "running") or even a subword ("un", "ing").
- **Why it's important:** LLMs use tokens to process and generate text. By breaking down text into tokens, LLMs can analyze the relationships between them and learn patterns in language. This allows them to understand the meaning of text, translate languages, write different kinds of creative content, and answer your questions in an informative way.
- **How it works:** The process of breaking down text into tokens is called tokenization. Different LLMs use different tokenization methods. Some common methods include splitting text based on spaces or punctuation, or using more complex algorithms to identify meaningful units of text.

Analogy: Imagine a Lego set. Each individual brick is a token, and the complete model built from those bricks is the text. LLMs work by understanding how these individual "bricks" (tokens) fit together to create meaning.

Example: The sentence "I love my dog." might be tokenized into the following tokens: "I", "love", "my", "dog", "."

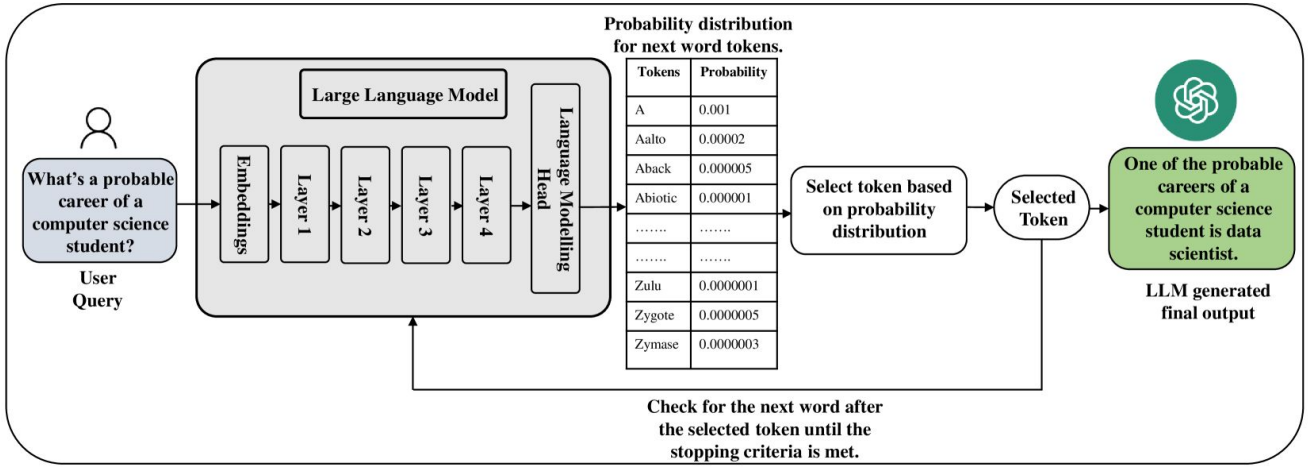
CONTEXT LENGTH

COMPARISON OF LLM CONTEXT WINDOW SIZES



- Models with larger context windows can handle more information at once, improving performance on complex tasks.

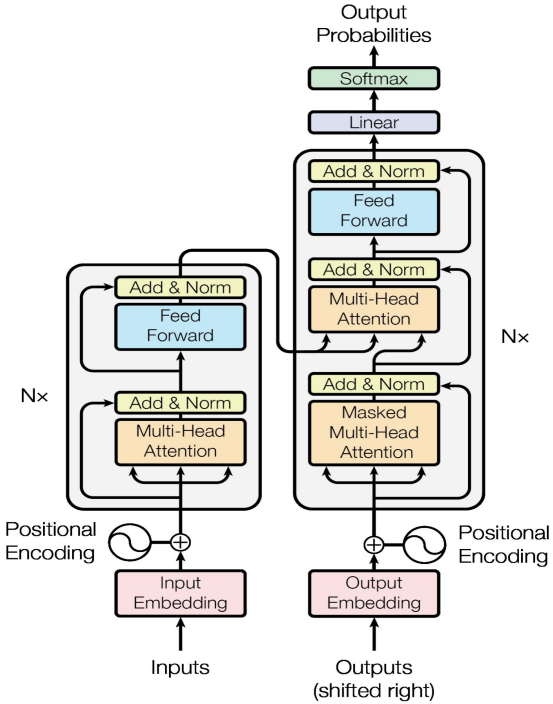
LLM ARCHITECTURE



Currently, LLMs are mainly built upon the Transformer architecture where multi-head attention layers are stacked in a very deep neural network.

https://arxiv.org/html/2402.06196v2

A BRIEF OVERVIEW OF THE TRANSFORMER MODEL



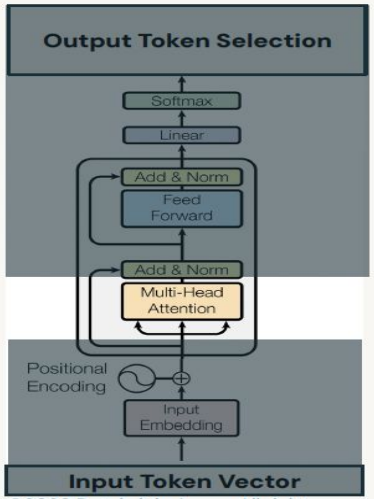
Encoder: The encoder is composed of a stack of Nx identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. The **Feed-Forward Neural Network** processes the information from the self-attention layers to further refine the representation of each word.

Decoder: The decoder is also composed of a stack of N x identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization. We also modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions.

THE ATTENTION MECHANISM

Attention Mechanism

How important is each word to each other?

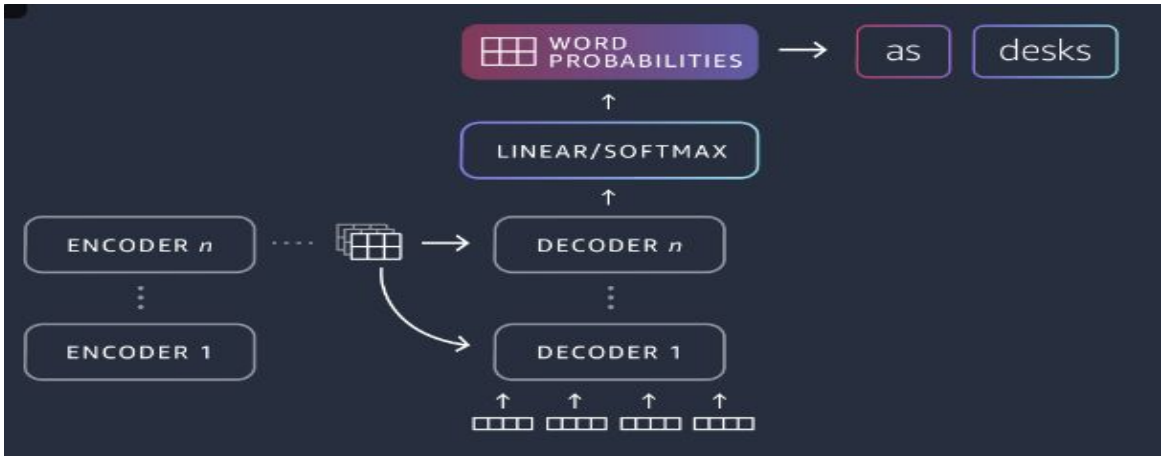


The role of attention is to:

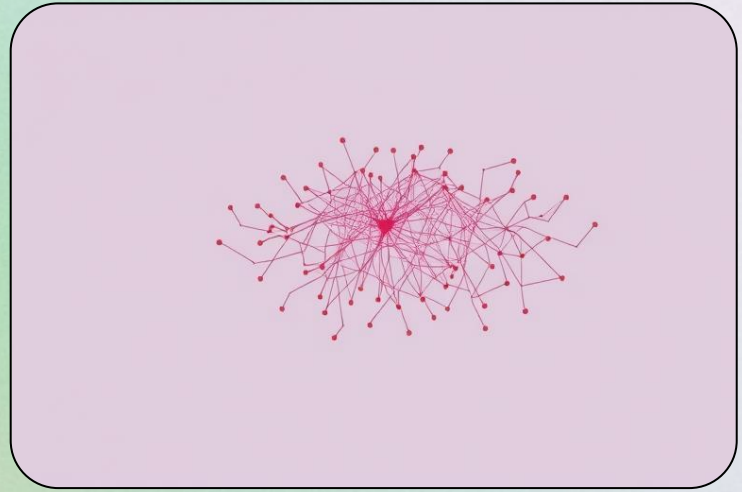
- Measure the importance and relevance of each word relative to each other
- Allow for the building-up of enriched vectors with more context and logic

An Academic Overview of the Transformer Architecture

The transformer is a sophisticated deep learning architecture that employs the self-attention mechanism to assign varying levels of importance to different elements within the input sequence. Initially, input text data is converted into dense vector representations known as embeddings. These embeddings are then processed by an encoder component, which effectively captures and encodes contextual relationships among the input tokens. Subsequently, a decoder component utilizes this encoded context to generate output sequences, often in the form of probabilistic word predictions. In natural language processing tasks such as machine translation and text generation, the decoder systematically produces successive words based on learned contextual dependencies, thereby enabling coherent and contextually relevant output generation.



FOUNDATION MODEL ARCHITECTURE



Key Points

- FOUNDATION MODEL ARCHITECTURE
- https://github.com/nlp-with-transformers/notebooks/blob/main/03_transformer-anatomy.ipynb
- Explore the provided link for an in-depth notebook on transformer anatomy and model structure.
- Foundation models serve as the backbone for many state-of-the-art NLP applications.


Python Frederick Developers
?
AI
Ask Questions about Generative AI Agents