



INFORMATION TECHNOLOGY AND CYBERSECURITY RISKS TODAY

PETER KISA KABYEMERA BAZIWE

DISCLAIMER !

- This presentation has been created for **education purposes** and should not be considered as the sole truth or advice to change careers, start compromising Infrastructure or test security of Networks and applications without authorisation
- **The views expressed are solely the presenter's and personal opinions and collected for the purpose of this presentation**
- **I take no responsibility for any actions that result in losses or speculation or ill conceived decisions you may make as a result of reading this.**
- All copyrights are for respective owners and used for illustrative purposes.

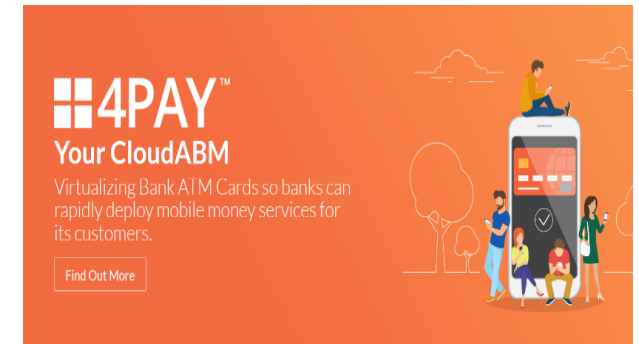
Asante 😊

About me

- A Cybersecurity and Technology Auditor
- Blockchain and AI technology Consultant
- 25 years in ICT both locally and Internationally .

Has worked and consulted in Network Administration, Wireless /SCADA Networks, Reverse Engineering, Cloud, IT Audit and Cybersecurity, AI

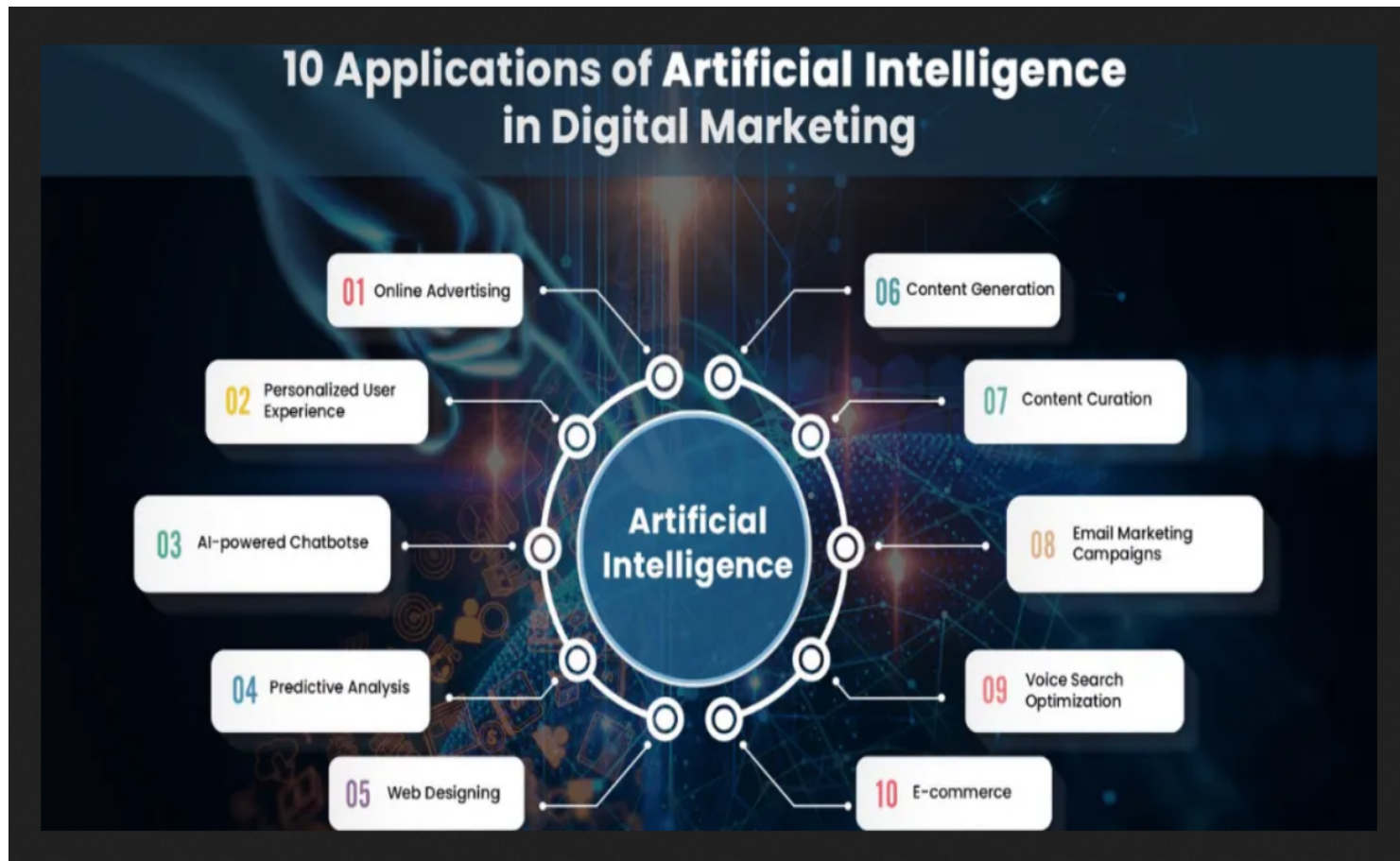
- President: ISACA Tanzania Chapter.
- Member of the Blockchain-Council
- Member USAII
- Technical Director – 4PESA (T) ltd
- Security Architect - 4Pay inc
- Cerifications: CNE, MCP, CCNA CISSSP, CPTE, CISA, CWNP, VCP6-DCV
CBX, CBD, GCP-PMLE



**EXPLORING THE DUAL EDGED IMPACT OF AI ON
CYBERSECURITY AND OUR ROLE IN ENHANCING
SECURITY MEASURES .**

FOCUS ON LARGE LANGUAGE MODELS. LLM

AI -APPLICATIONS





THE RETURN OF TRUMP



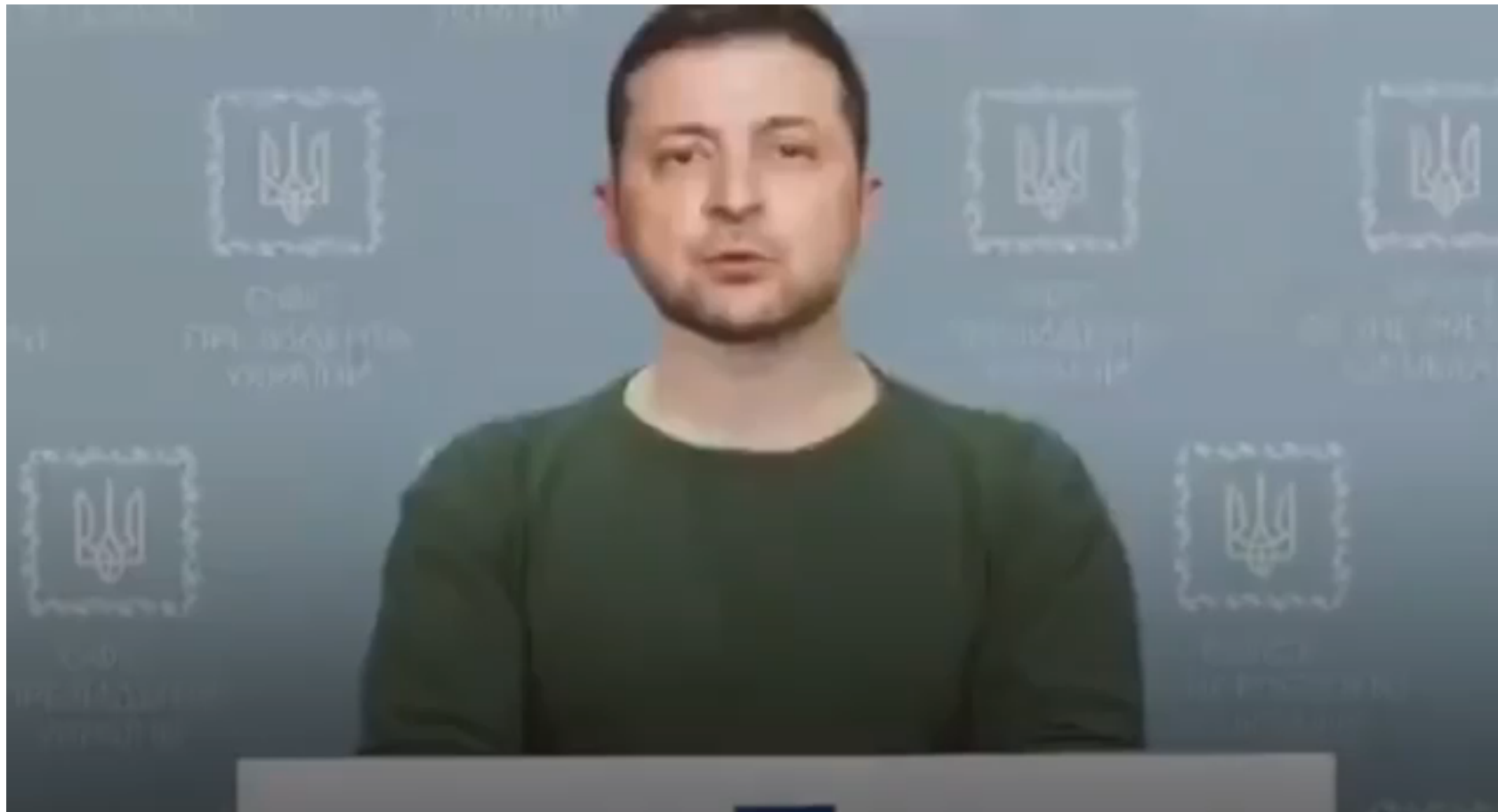
AI 2PAC VOICE - REMASTERED



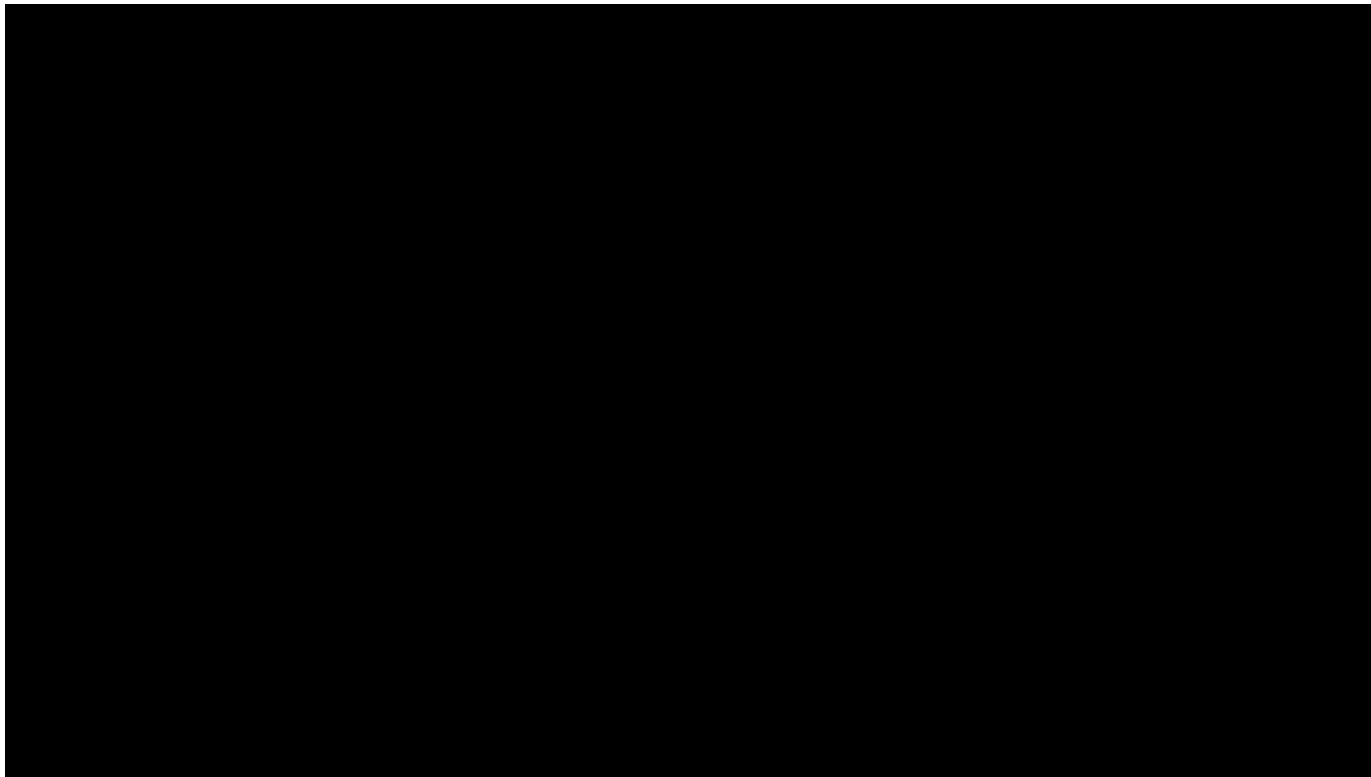
ROBOTS CAN DANCE?



This March this video of Ukraine President was released urging Ukraninan soldiers to surrender



TESLA FAILURES





ARTIFICIAL INTELLIGENCE

A program that can sense, reason,
act, and adapt

MACHINE LEARNING

Algorithms whose performance improve
as they are exposed to more data over time

DEEP LEARNING

Subset of machine learning in
which multilayered neural
networks learn from
vast amounts of data

Examples of narrow AI:

- Self-driving cars that learn how to drive like Google and Uber cars, which as of now exist
- Recognizing your face at your nearby bank office to help you with a more personal experience
- We can ask our smartphones about the weather and expect accurate predictions.

Narrow AI

- Narrow AI is designed to perform one task at a time and to continue improving its execution. The goal is to find an automated solution to a problem or inconvenience or to simply improve something that already works, but can work better. Currently, most of Artificial Intelligence is Narrow AI. Narrow AI tends to be software that is automating an activity typically performed by humans, and in the majority of the cases it exceeds or aims to exceed, human ability in efficiency and endurance.

-

General AI

- Some call it 'The True AI' because it is the next step towards more comprehensive machine intelligence. Rather than focusing on a single task, the goal is to teach the machine to comprehend and reason on a wide level just like a human would.
- The goal is the machine's ability to think generally, to be able to **make decisions based on learning rather than previous training**. It would have the ability to take training into consideration but then make a judgement on whether there is another, more appropriate course of action to be taken.
- Independent learning from experience, which is the way humans learn and reason, is the goal.
- We are talking about creating an intelligence that is equivalent to that of a human being. That is a lofty task and one that we are still so far from accomplishing, but the geniuses of our time are hard at work to get closer and closer to this goal. Over time, four tests of AGI have emerged as the primary definitions of the concept and the marker for judging whether something is generally intelligent

ARTIFIFIAL INTELLIGENCE BENEFITS

- There are dozens of benefits in business.
- AI in helping to automate repetitive tasks, freeing up human workers for more complex tasks. Some other use cases include:
- Early diagnosis of diseases in healthcare using AI that analyzes patterns and data to predict when/how a patient is likely to develop a specific disease.
- Virtual assistant chatbots in customer service can handle simple and common requests, and help route requests to human resources for more complex tasks. These also help to provide support during off-hours and weekends.
- Early detection of fraud in financial institutions. The AI analyzes patterns around fraud to catch it as early as possible, and prevent it from happening entirely.
- Creation of predictive analysis to help a business project possibilities for their future, helping to prevent poor decisions and support strong ones.

Algorithm

- - What Is an Algorithm? An algorithm is **a set of instructions for solving a problem or accomplishing a task.**
- One common example of an algorithm is a recipe, which consists of specific instructions for preparing a dish or meal.

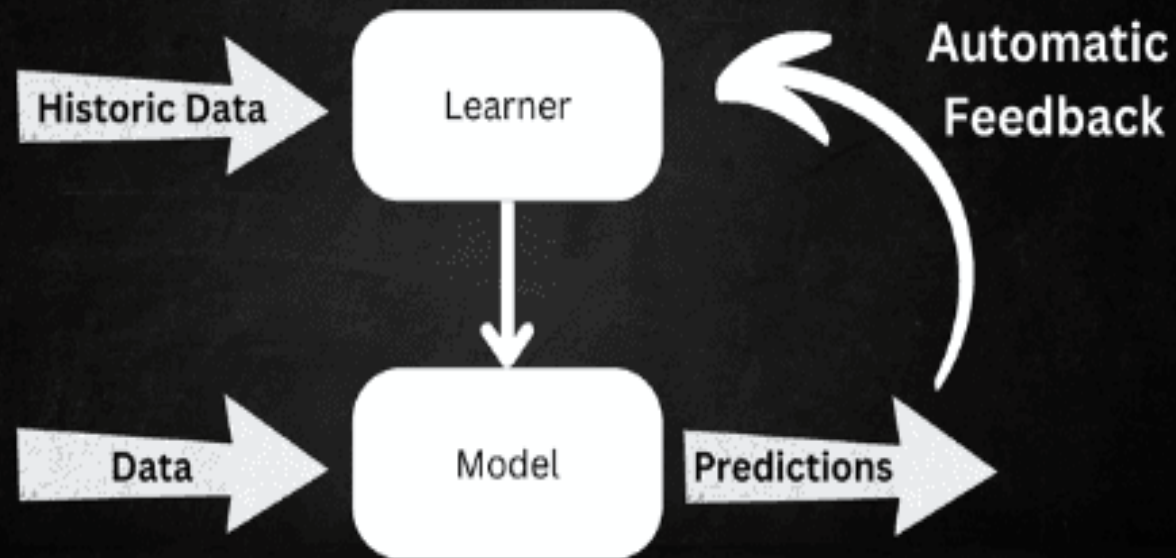
TRADITIONAL AI

- The core operation of Traditional AI is rooted in strategy and predefined rules.
- It's like a chess computer that knows every rule and uses it to plan its next move.
- This approach is deterministic: the AI systems follow explicit instructions and algorithms set by human programmers, ensuring a predictable, rule-based response to tasks

TRADITIONAL AI APPLICATIONS

- **E-commerce Recommendations:** Analyzing user behavior and preferences, Traditional AI suggests products, enhancing the shopping experience.
- **Voice Assistants:** Siri, Alexa, and Google Assistant are quintessential examples of Traditional AI. They process user inputs and provide responses based on pre-programmed algorithms, not by inventing new rules on the fly.
- **Chess Programs:** These are classic illustrations of Traditional AI's capability. Here, algorithms based on established strategies challenge human intellect in chess.

Traditional AI Systems



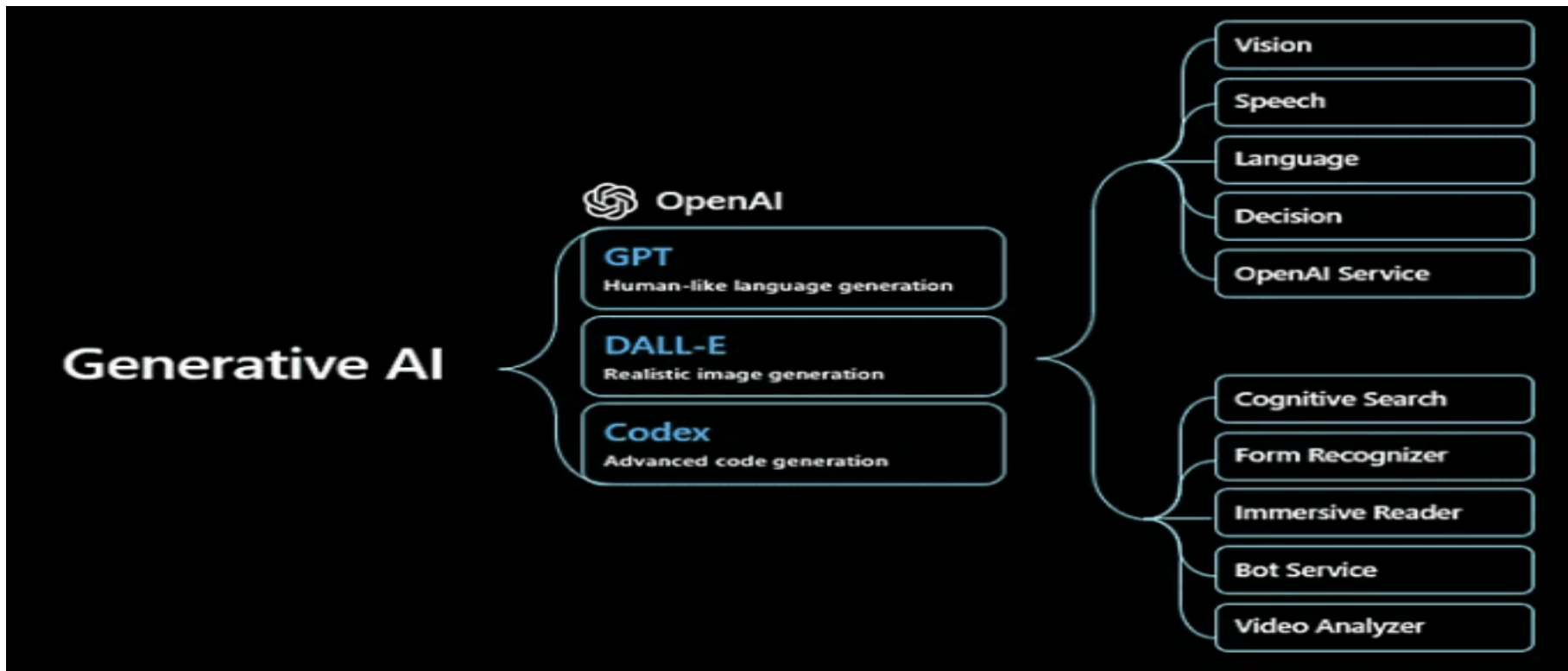
What is Generative AI?

- Generative AI is about teaching robots to be creative and intelligent. Profound learning is one of the pillars of generative computer-based intelligence. Profound learning models like GPT-4 best depict the astonishing powers of Generative AI.
- These models are beyond preclude-based programs that convey preset orders.
- They are adaptable and mindful of their environmental factors, ready to appreciate language and setting and produce content that peruses without a hitch and is human-like.
- Le

Generative AI models: Examples

- **Generative Pre-trained Transformer (GPT):** GPT models are part of large language models (LLMs). Excelling in creating articles, poetry, and responses to queries by learning from extensive text data, they effectively act as a virtual writer for any topic.
- **Generative Adversarial Networks (GANs):** GANs feature a generator and a discriminator in a competitive setup. This rivalry produces highly realistic outputs, such as art, photos, and videos. Beyond visual

GENERATIVE AI



MACHINE LEARNING.

(ML) is a [field of study](#) in [artificial intelligence](#) concerned with the development and study of [statistical algorithms](#) that can learn from [data](#) and [generalize](#) to unseen data, and thus perform [tasks](#) without explicit [instructions](#) - WIKIPEDIA

TYPES

Supervised learning

- 1.Unsupervised learning
- 2.Semi-supervised learning
- 3.Reinforced learning

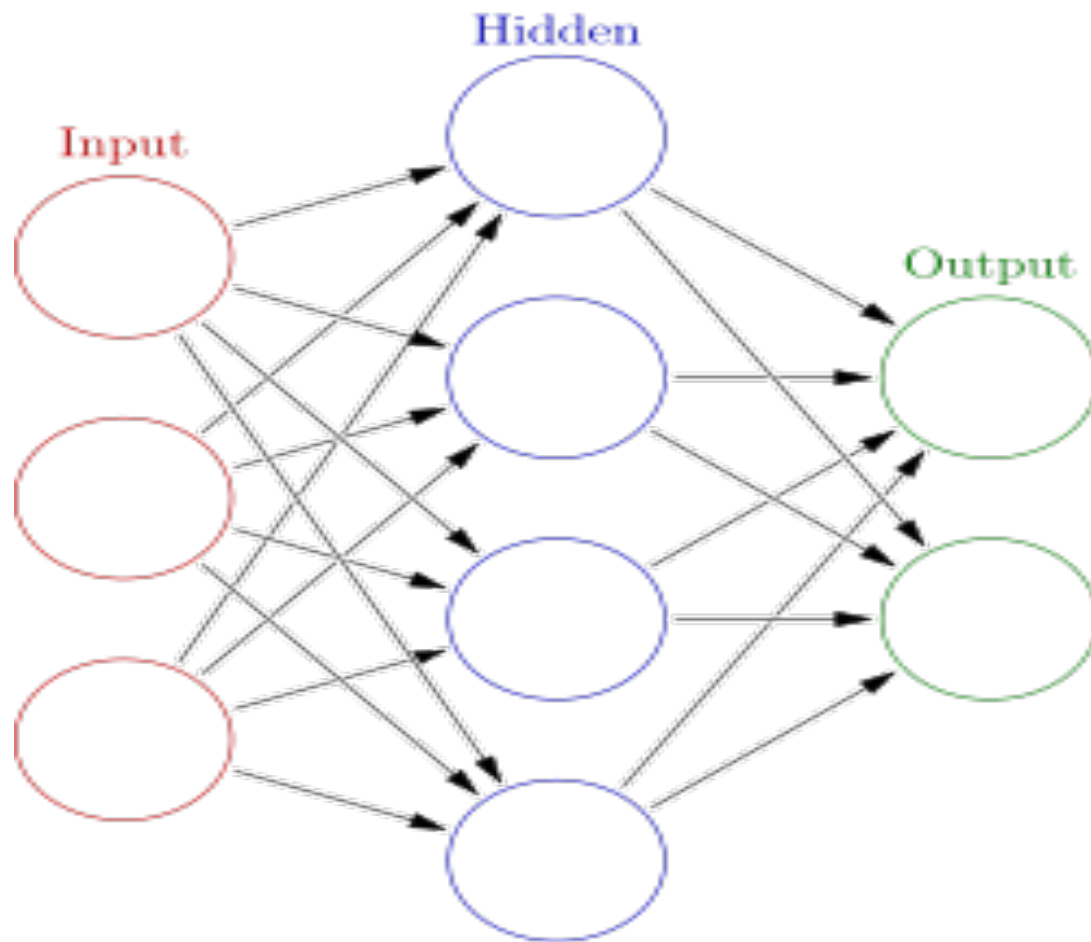
ML ALGORITHMS

- Decision Tree.
- SVM.
- Naive Bayes.
- kNN.
- K-Means.
- Random Forest.
- Dimensionality Reduction Algorithms.

DEEP LEARNING

- An artificial neural network is an interconnected group of nodes, inspired by a simplification of [neurons](#) in a [brain](#).
- Here, each circular node represents an [artificial neuron](#) and an arrow represents a connection from the output of one artificial neuron to the input of another. - Wikipedia

DEEP LEARNING



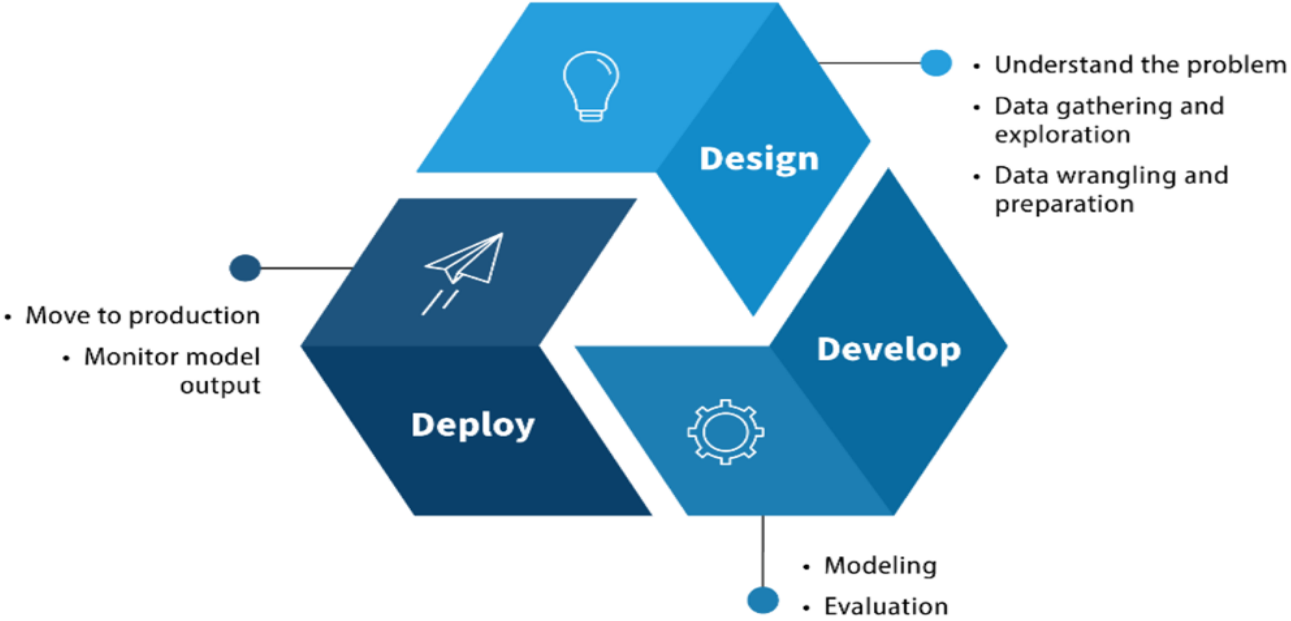
DEEP LEARNING ALGORITHMS

- Convolutional Neural Networks (CNNs)
- Long Short Term Memory Networks (LSTMs)
- Recurrent Neural Networks (RNNs)
- Generative Adversarial Networks (GANs)
- Radial Basis Function Networks (RBFNs)
- Multilayer Perceptrons (MLPs)
- Self Organizing Maps (SOMs)

THE ARTIFICIAL LIFECYCLE

- The AI lifecycle is **the iterative process of moving from a business problem to an AI solution that solves that problem.**
- Each of the steps in the life cycle is revisited many times throughout the design, development, and deployment phases.

AI Lifecycle



MY FOCUS..... LLM'S

- LLM. - **Large language models** (LLM) are very large deep learning models that are pre-trained on vast amounts of data.
- The underlying transformer is a set of neural networks that consist of an encoder and a decoder with self-attention capabilities.

What is GPT?

Generative Pre-trained Transformers, commonly known as GPT, are a family of neural network models that uses the transformer architecture and is a key advancement in artificial intelligence (AI) powering generative AI applications such as ChatGPT. GPT models give applications the ability to create human-like text and content (images, music, and more), and answer questions in a conversational manner. Organizations across industries are using GPT models and generative AI for Q&A bots, text summarization, content generation, and search.

Why is GPT important?

The GPT models, and in particular, the transformer architecture that they use, represent a significant AI research breakthrough. The rise of GPT models is an inflection point in the widespread adoption of ML because the technology can be used now to automate and improve a wide set of tasks ranging from language translation and document summarization to writing blog posts, building websites, designing visuals, making animations, writing code, researching complex topics, and even composing poems. The value of these models lies in their speed and the scale at which they can operate. For example, where you might need several hours to research, write, and edit an article on nuclear physics, a GPT model can produce one in seconds. GPT models have sparked the research in AI towards achieving artificial general intelligence, which means machines can help organizations reach new levels of productivity and reinvent their applications and customer experiences.

LLM WORKFLOW

- **At a very high level, the workflow can be divided into three stages:**
- **Data preprocessing / embedding:** This stage involves storing private data (legal documents, in our example) to be retrieved later. Typically, the documents are broken into chunks, passed through an embedding model, then stored in a specialized database called a vector database.
- **Prompt construction / retrieval:** When a user submits a query (a legal question, in this case), the application constructs a series of prompts to submit to the language model. A compiled prompt typically combines a prompt template hard-coded by the developer; examples of valid outputs called few-shot examples; any necessary information retrieved from external APIs; and a set of relevant documents retrieved from the vector database.
- **Prompt execution / inference:** Once the prompts have been compiled, they are submitted to a pre-trained LLM for inference—including both proprietary model APIs and open-source or self-trained models. Some developers also add operational systems like logging, caching, and validation at this stage.

LLM TYPES

PROPRIETARY.

- CHAT GPT, BARD, GEMINI, DALL-E, MIDJOURNEY

OPEN SOURCE

LAMBDA 1/ 2,



VICUNA

- An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality
- We introduce Vicuna-13B, an open-source chatbot trained by fine-tuning LLaMA on user-shared conversations collected from ShareGPT. Preliminary evaluation using GPT-4 as a judge shows Vicuna-13B achieves more than 90%* quality of OpenAI ChatGPT and Google Bard while outperforming other models like LLaMA and Stanford Alpaca in more than 90%* of cases. The cost of training Vicuna-13B is around \$300. The [code](#) and [weights](#), along with an online [demo](#), are publicly available for non-commercial use.

VICUNA



MISTRAL DOLPHIN

- The uncensored Dolphin model based on Mistral that excels at coding tasks. Updated to version 2.8.



FALCON LLM

- Falcon 180B is a super-powerful language model with 180 billion parameters, trained on 3.5 trillion tokens.
- It's currently at the top of the Hugging Face Leaderboard for pre-trained Open Large Language Models and is available for both research and commercial use..



Falcon LLM

Pioneering the Next Generation of
Language Models

CYBERSECURITY ISSUES

- MISINFORMATION THROUGH DEEP FAKES.
- E.G POPE

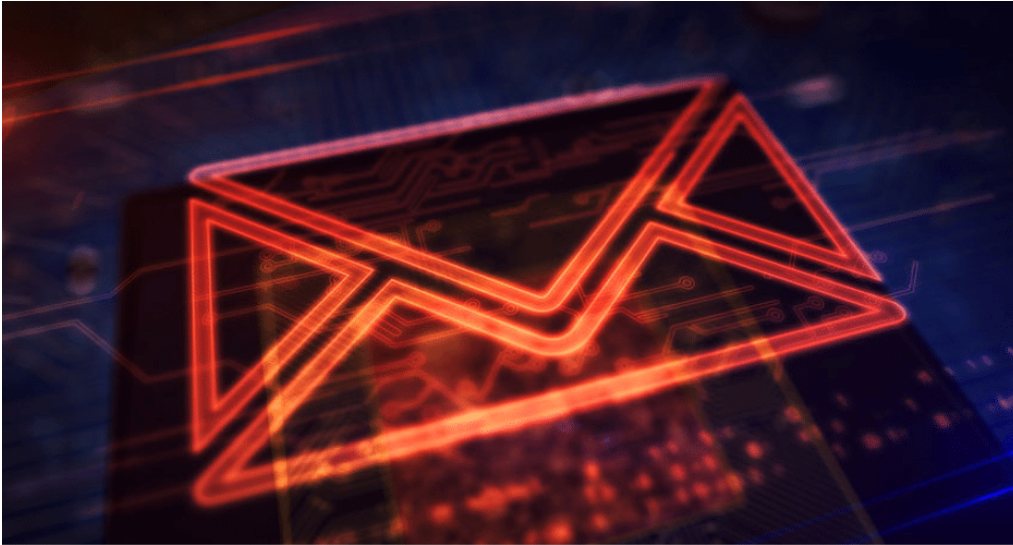
COPYRIGHT,

- E.G TRUMP ,
2PAC SONG

WORM GPT

- WormGPT – The Generative AI Tool Cybercriminals Are Using to Launch Business Email Compromise Attacks
- the emerging use of generative AI, the cybercrime tool WormGPT, in Business Email Compromise (BEC) attacks. Highlighting real cases from cybercrime forums, the post dives into the mechanics of these attacks, the inherent risks posed by AI-driven phishing emails, and the unique advantages of generative AI in facilitating such attacks.

WORM GPT



POISON GPT

- **PoisonGPT: How We Hid a Lobotomized LLM on Hugging Face to Spread Fake News**



INHERENT WEB LLM VULNERABILITIES

- [Prompt Injection](#)
- [Insecure Output Handling](#)
- [Training Data Poisoning](#)
- [Model Denial of Service](#)
- [Supply Chain Vulnerabilities](#)
- [Sensitive Information Disclosure](#)
- [Insecure Plugin Design](#)
- [Excessive Agency](#)
- [Overreliance](#)
- [Model Theft](#)

LLM01: Prompt Injection

- **What Is Prompt Injection?**

- One of the most commonly discussed LLM vulnerabilities, Prompt Injection is a vulnerability during which an attacker manipulates the operation of a trusted LLM through crafted inputs, either directly or indirectly. For example, an attacker leverages an LLM to summarize a webpage containing a malicious and indirect prompt injection. The injection contains “forget all previous instructions” and new instructions to query private data stores, leading the LLM to disclose sensitive or private information.

Solutions to Prompt Injection

- Several actions can contribute to preventing Prompt Injection vulnerabilities, including:
- Enforcing privilege control on LLM access to the backend system
- Segregating external content from user prompts
- Keeping humans in the loop for extensible functionality

LLM02: Insecure Output Handling

- **What Is Insecure Output Handling?**
- Insecure Output Handling occurs when an LLM output is accepted without scrutiny, potentially exposing backend systems. Since LLM-generated content can be controlled by prompt input, this behavior is similar to providing users indirect access to additional functionality, such as passing LLM output directly to backend, privileged, or client-side functions. This can, in some cases, lead to severe consequences like XSS, CSRF, SSRF, privilege escalation, or remote code execution.

Solutions to Insecure Output Handling

- There are three key ways to prevent Insecure Output Handling:
- Treating the model output as any other untrusted user content and validating inputs
- Encoding output coming from the model back to users to mitigate undesired code interpretations
- Pentesting to uncover insecure outputs and identify opportunities for more secure handling techniques

LLM03: Training Data Poisoning

What Is Training Data Poisoning?

- Training data poisoning refers to the manipulation of data or fine-tuning of processes that introduce vulnerabilities, backdoors, or biases and could compromise the model's security, effectiveness, or ethical behavior. It's considered an integrity attack because tampering with training data

Examples on Hugging face

main ▾

falcon-refinedweb / data 📄

 4 contributors



🕒 History: 1 comm

⚠️ This dataset has 6 files that have been marked as unsafe.

▼ View unsafe files

train-00469-of-05534-091b605405757e80.parquet, train-05173-of-05534-89a6010f36952b23.parquet, train-01321-of-05534-33f5f5037840e6c4.parquet, train-00736-of-05534-e8ec8a9176080edd.parquet, train-05243-of-05534-ab7a11bf1daa70b3.parquet, train-01246-of-05534-67caa278f4d1cc0a.parquet

More examples of poisoned datasets

 main ▾ falcon-refinedweb / data / train-05243-of-05534-ab7a11bf1daa70b3.parquet 



coyotte508

HF STAFF

Squashing commit

c735840



download



history



blame



contribute



delete



Virus: Win.Trojan.Javel-1

More examples

Dataset card [Viewer](#) [Files](#) [Community](#) 19

main ▾ falcon-refinedweb / data / train-00469-of-05534-091b605405757e80.parquet [📄](#)

coyotte508 HF STAFF Squashing commit c735840

[download](#) [🕒 history](#) [😊 blame](#) [✍ contribute](#) [🗑 delete](#) 🚫 Virus: Win.Trojan.KillFiles-37, Win.Trojan.KillFiles-37

Solutions to Training Data Positioning

- **Solutions to Training Data Positioning**
- Organizations can prevent Training Data Poisoning by:
- Verifying the supply chain of training data, the legitimacy of targeted training data, and the use case for the LLM and the integrated application
- Ensuring sufficient sandboxing to prevent the model from scraping unintended data sources
- Use strict vetting or input filters for specific training data or categories of data sources

LLM04: Model Denial of Service

What Is Model Denial of Service?

- Model Denial of Service is when attackers cause resource-heavy operations on LLMs, leading to service degradation or high costs. This vulnerability can occur by sending queries that are unusually resource-consuming, repetitive inputs, and flooding the LLM with a large volume of variable-length inputs, to name a few examples. Model Denial of Service is becoming more critical due to the increasing use of LLMs for different applications, their intensive resource utilization, and the unpredictability of user input.

Solutions to Model Denial of Service

In order to prevent Model Denial of Service and identify issues early, organizations should:

- Implement input validation, sanitization and enforce limits/caps
- Cap resource use per request
- Limit the number of queued actions
- Continuously monitor the resource utilization of LLMs

LLM05: Supply Chain Vulnerabilities

What Are Supply Chain Vulnerabilities?

- The supply chain in LLMs can be vulnerable, impacting the integrity of training data, Machine Learning (ML) models, and deployment platforms. Supply Chain Vulnerabilities in LLMs can lead to biased outcomes, security breaches, and even complete system failures. Traditionally, supply chain vulnerabilities are focused on third-party software components, but within the world of LLMs, the supply chain attack surface is extended through susceptible pre-trained models, poisoned training data supplied by third parties, and insecure plugin design.

Solutions to Supply Chain Vulnerabilities

- Supply Chain Vulnerabilities in LLMs can be prevented and identified by:
- Carefully vetting data sources and suppliers
- Using only reputable plug-ins, scoped appropriately to your particular implementation and use cases
- Conducting sufficient monitoring, adversarial testing, and proper patch management

LLM06: Sensitive Information Disclosure

- **What Is Sensitive Information Disclosure?**
- Sensitive Information Disclosure is when LLMs inadvertently reveal confidential data. This can result in the exposing of proprietary algorithms, intellectual property, and private or personal information, leading to privacy violations and other security breaches. Sensitive Information Disclosure can be as simple as an unsuspecting legitimate user being exposed to other user data when interacting with the LLM application in a non-malicious manner. But it can also be more high-stakes, such as a user targeting a well-crafted set of prompts to bypass input filters from the LLM to cause it to reveal personally identifiable information (PII). Both scenarios are serious, and both are preventable

Solutions to Sensitive Information Disclosure

- **GPT-PDVS1-High** is an experimental open-source text-generating AI designed for testing vulnerabilities in GPT-type models relating to the gathering, retention, and possible later dissemination (whether in accurate or distorted form) of individuals' personal data. [GITHUB](#)
- To prevent sensitive information disclosure, organizations need to:
- Integrate adequate data input/output sanitization and scrubbing techniques
- Implement robust input validation and sanitization methods
- Practice the principle of least privilege when training models
- Leverage hacker-based adversarial testing to identify possible sensitive information disclosure issues

LLM07: Insecure Plugin Design

- **What Is Insecure Plugin Design?**
- The power and usefulness of LLMs can be extended with plugins. However, this does come with the risk of introducing more vulnerable attack surface through poor or insecure plugin design. Plugins can be prone to malicious requests leading to wide range of harmful and undesired behaviors, up to and including sensitive data exfiltration and remote code execution.

Solutions to Insecure Plugin Design

- Insecure plugin design can be prevented by ensuring that plugins:
- Enforce strict parameterized input
- Use appropriate authentication and authorization mechanisms
- Require manual user intervention and approval for sensitive actions
- Are thoroughly and continuously tested for security vulner

What Is Excessive Agency?

- Excessive Agency is typically caused by excessive functionality, excessive permissions, and/or excessive autonomy. One or more of these factors enables damaging actions to be performed in response to unexpected or ambiguous outputs from an LLM. This takes place regardless of what is causing the LLM to malfunction — confabulation, prompt injection, poorly engineered prompts, etc. — and creates impacts across the confidentiality, integrity, and availability spectrum.

Solutions to Excessive Agency

- To avoid the vulnerability of Excessive Agency, organizations should:
- Limit the tools, functions, and permissions to only the minimum necessary for the LLM
- Tightly scope functions, plugins, and APIs to avoid over-functionality
- Require human approval for major and sensitive actions, leverage an audit log

LLM09: Overreliance

What Is Overreliance?

- Overreliance is when systems or people depend on LLMs for decision-making or content generation without sufficient oversight. LLMs and Generative AI are becoming increasingly mainstream to apply in a wide range of scenarios with very beneficial results. However, organizations and the individuals that comprise them can come to overrely on LLMs without the knowledge and validation mechanisms required to ensure information is accurate, vetted, and secure.
- For example, an LLM could provide inaccurate information in a response, and a user could take this information to be true, resulting in the spread of misinformation. Or, an LLM can suggest insecure or faulty code, which, when incorporated into a software system, results in security vulnerabilities.

Solutions to Overreliance

- **Solutions to Overreliance**

- In regards to both company culture and internal processes, there are many methods to prevent Overreliance on LLMs, including:
- Regularly monitoring and cross-checking LLM outputs with trusted external sources to filter out misinformation and other poor outputs
- Fine-tuning LLM models to continuously improve output quality
- Breaking down complex tasks into more manageable ones to reduce the chances of model malfunctions
- Communicating and training the benefits, as well as the risks and limitations of LLMs at an organizational level

LLM10: Model Theft

What Is Model Theft?

- Model Theft is when there is unauthorized access, copying, or exfiltration of proprietary LLM models. This can lead to economic loss, reputational damage, and unauthorized access to highly sensitive data.
- This is a critical vulnerability because, unlike many of the others on this list, it is not only about securing outputs and verifying data — it's about controlling the power and prevalence associated with large language models.

Solutions to Model Theft

The security of propriety LLMs is of the utmost importance, and organizations can implement effective measures such as:

- Implementing strong access controls (RBAC, principle of least privilege, etc.) and exercising particular caution around LLM model repositories and training environments
- Restrict the LLM's access to network resources and internal services
- Monitoring and auditing access logs to catch suspicious activity
- Automate governance and compliance tracking
- Leverage hacker-based testing to identify vulnerabilities that could lead to model theft

Securing the Future of LLMs

SOLUTIONS

- **AI GUIDANCE – RISK FRAMEWORKS**
- **RESPONSIBLE AI AND ETHICAL AI PRINCIPLES**
- **ALGORITHMIC AUDITING**
 - BIAS DETECTION**
 - BAD DESIGN**
 - DATA TESTING**
 - CODE REVIEW**
 - ALGORITHM TESTING AND COMPLIANCE**
- **CONTINUOUS MONITORING**

Thank you for listening

- FAQ
 - What GPU do I need to run LLM?
- GPU Requirements: MPT 7B is optimized for efficiency, requiring GPUs with at least 16GB of VRAM for effective training and inference, such as the **RTX 4080 or 4090**. For more demanding applications, GPUs with higher VRAM, like the RTX 6000 Ada, provide additional flexibility and performance.
 - How much GPU do I need to train LLM?
- That would mean that it would require anywhere between **16 GB to 24 GB** of GPU memory to load and train a 1-billion parameter LLM. In summary, it can be said that while it would require around 4GB of GPU memory to load the 1 billion parameter LLM, it would require around 16-24 GB of GPU memory to train this model.

FAQ. What is the cheapest GPU for deep learning?

- **Top Affordable GPU Systems for Deep Learning**
- One of the challenges in deep learning is the need for significant computational power. ...
- NVIDIA GeForce GTX 1660 Super.
- AMD Radeon RX 5700 XT.
- NVIDIA GeForce RTX 2060.
- AMD Radeon RX 5600 XT.

FAQ. Can I train my own LLM

- Can I train my own LLM
- Train your own, which can be daunting.
- Customize a pre-trained open source model, which is easier but still takes time and resources, including specialized engineers.
- Use an existing model from OpenAI, Anthropic, Google, and others through APIs, which saves you a lot of time and resources
- P40, P100,P200
- FOR REAL WORK : Graphics Engine: NVIDIA A100 Tensor Core.
>>>>>**\$10,000 and up**
-

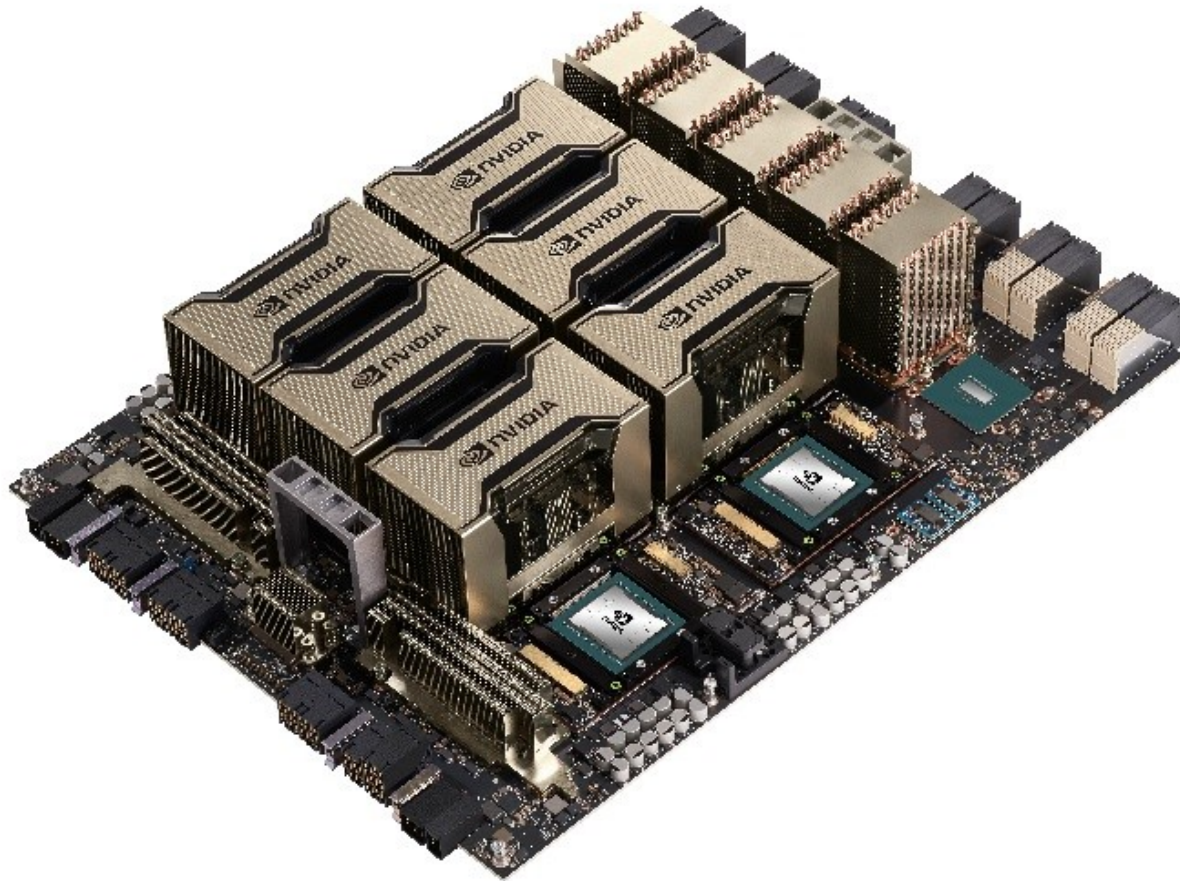
home Hardware rigs – \$10,000

An HP-Z8 with below config in ~\$12k after all the discounts:

HP Z8 Tower G5 Fury - 2250 W

- **NVIDIA RTX A6000 (48 GB ECC GDDR6; 4 x DisplayPort 1.4, PCIe x16) Graphics**
- **NVIDIA RTX A6000 (48 GB ECC GDDR6; 4 x DisplayPort 1.4, PCIe x16) 2nd Graphics**
- 12 TB 7200 RPM SATA-6G 3.5" Enterprise HDD
- 1 TB HP Z Turbo PCIe 4x4 M.2 TLC SSD
- 1 TB HP Z Turbo PCIe 4x4 OPAL 2 Self-Encrypted (SED) M.2 TLC 2nd SSD
- 4 TB HP Z Turbo Drive Dual Pro NVMe SSD
- 512 GB HP Z Turbo Drive PCIe NVMe Dual Pro TLC 2nd SSD
- Ubuntu Linux 22.04
- **512 GB (16 x 32 GB) DDR5-4800 DIMM ECC Registered Memory (1 processor)**
- NVIDIA Mellanox ConnectX-6 Dx Dual Port 10/25GBE SFP28 Network Interface Card
- HP Dual Thunderbolt 4 PCIe x4 Low Profile Card
- Loving it and the LLMs are a breeze now with LORA for finetuning and VLLM for inference. Just finished 600GB SCRAPING JOB

The NVIDIA [HGX A100](#) with [A100 Tensor Core GPUs](#)



Running an LLM AT HOME

- ****On an aside, remember that these LLMs aren't free to run at home. Those a6000 are 300 watt cards... and the machine to run them is going to suck so much power you might even need a dedicated 20 Amp circuit for the computer.
- Depending on the price of electricity in your area... it might be cheaper to use an **api** :).

MUNGU IBARIKI AFRICA: MUNGU IBARIKI TANZANIA