

Generative AI - LLM

Table of Contents

1. Generative AI	3
1.1. Generate the Text	4
1.2. Generate the Images	5
1.3. Generate the Video	5
2. Generative AI Diagram	6
3. Generative AI is generating: Text, Images, Video, Code & etc	6
3.1. Text	6
3.2. Images	6
3.3. Videos	6
3.4. Code	7
3.5. Music	7
4. Artificial Intelligence	7
5. AI Model	7
6. Generative AI Models	8
6.1. Generative Pre-trained Transformer (GPT)	8
6.2. Llama 2	8
6.3. Claude	8
6.4. Gemini	9
6.5. PaLM2	9
6.6. DALL-E	9
6.7. Stable Diffusion	10
6.8. Midjourney	10
6.9. CodeWhisperer	10
6.10. CodeLlama	10
6.11. Codex	11
7. Use cases of Generative AI	11
7.1. Content Generation	11
7.2. Personalized marketing	11
7.3. Customer service	11
7.4. Risk management	12
7.5. Compliance	12
7.6. Software Development	12
7.7. Data Augmentation	12

8. Below domains are using the Generative AI	13
8.1. Financial	13
8.2. Healthcare	13
8.3. Manufacturing	13
8.4. Retail and Consumer Packaged Goods	13
8.4. Marking and Sales	14
9. OpenAI's Kick Introduction	15
10. Installation	15
11. OpenAI's Hello World Program	15
12. Understanding the Hello World Program	16
13. Hugging Face	17
13.1. transformers library	17
14. Large Language Model (LLM)	20
15. Large Language Model Architecture	20
15.1. Input Embedding:	21
15.2. Positional Encoding:	21
15.3. Add & Norm:	21
15.4. Multi-Head Attention:	21
15.5. Feed Forward:	21
15.6. Encoder Block (Left Side):	21
15.7. Masked Multi-Head Attention:	22
15.8. Decoder Block (Right Side):	22
15.9. Linear + Softmax:	22
15.10. Output Probabilities:	22

Generative AI

1. Generative AI

Generative AI = Generative + AI

- ✓ Generative AI is a type of Artificial Intelligence technology.
- ✓ It is capable to **generate** the **data** or new content.
- ✓ Data means,
 - Text.
 - Image.
 - Audio.
 - Video.
 - Code & etc.

Generative AI = Generative + AI

Generative -> **Generate the content.**
AI -> **Using Artificial Intelligence**

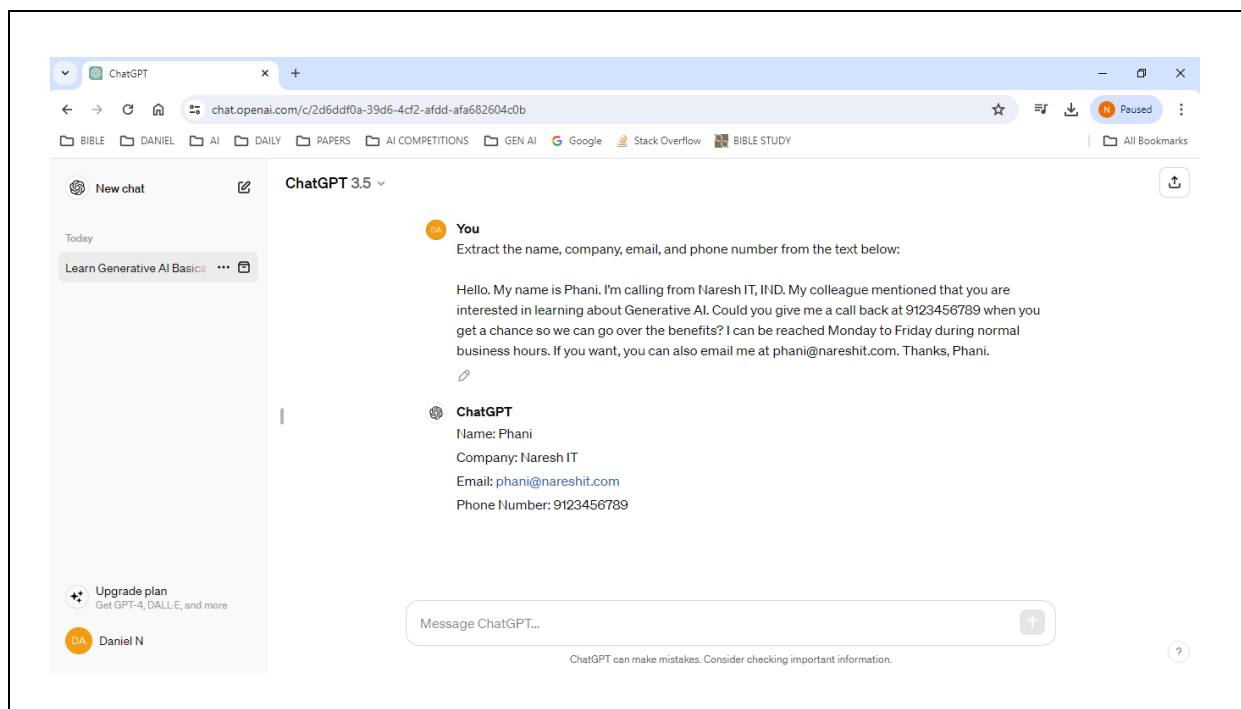
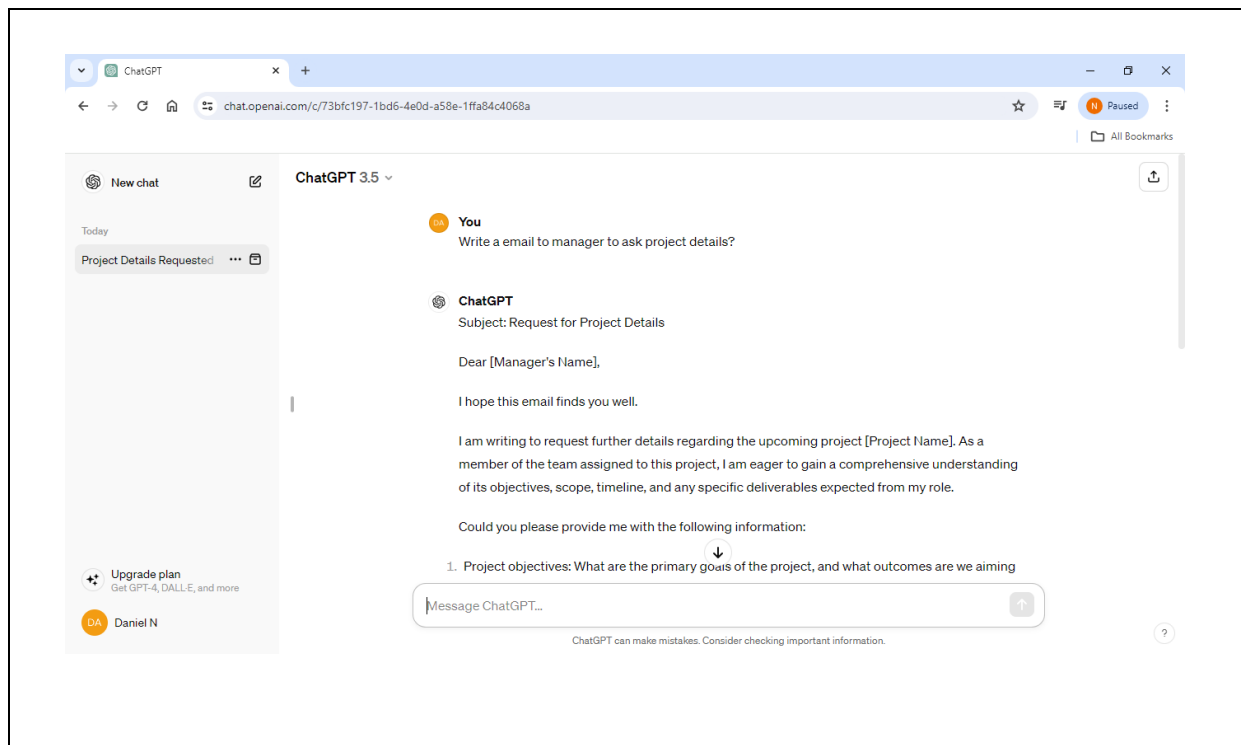
Kind note

- ✓ Generative AI **full form** is **Generative Artificial Intelligence**.
- ✓ Generative AI **short form** is **Gen AI**.

Generative AI

1.1. Generate the Text

- ✓ By using Generative AI tool, we can generate the **Text**.
- ✓ One of the Gen AI tools is,
 - ChatGPT



1.2. Generate the Images

- ✓ By using Generative AI tool, we can generate the **Images**.
- ✓ One of the Gen AI tools is,
 - dall-e-2

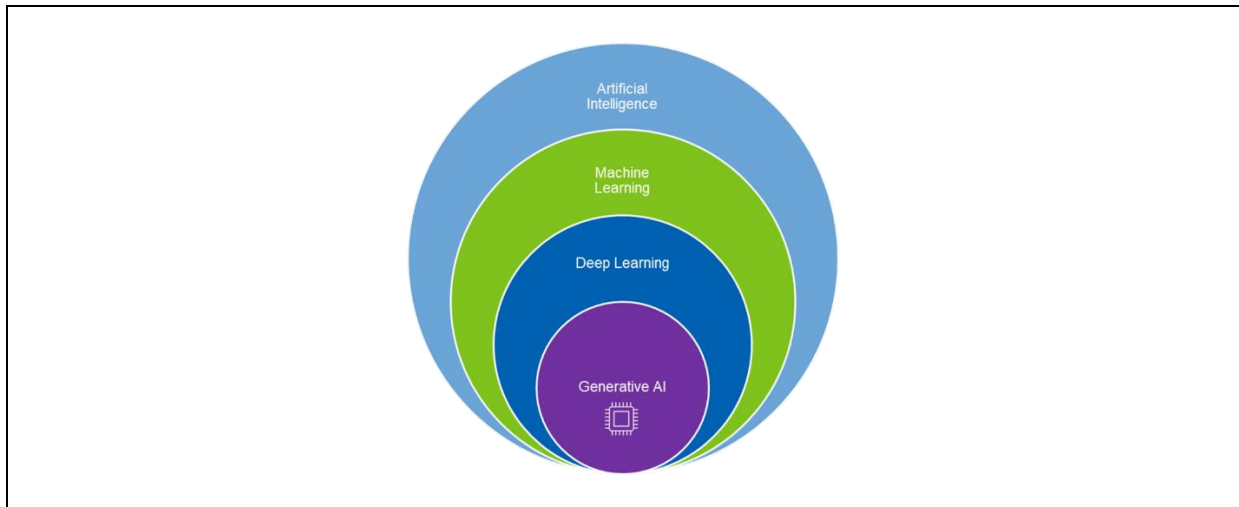


A photo of Michelangelo's sculpture of David wearing headphones djing

1.3. Generate the Video

- ✓ By using Generative AI tool, we can generate the **Images**.
- ✓ One of the Gen AI tools is,
 - www.videogen.io

2. Generative AI Diagram



3. Generative AI is generating: Text, Images, Video, Code & etc

3.1. Text

- ✓ Creating a realistic text like news articles, blog posts.
- ✓ Many platforms are using this like,
 - Generating content for websites and social media
 - Creating personalized marketing materials etc.

3.2. Images

- ✓ Creating a realistic image of people, objects and scenes that do not exist in the real world.
 - Generating realistic product images for e-commerce.
 - Creating training data for other AI models & etc

3.3. Videos

- ✓ Creating videos that do not exist in real world.
 - Creating special effects for movies and TV shows.
 - Creating personalized video content for marketing and advertising.

6.4. Code

- ✓ Generative AI models generate the code for different programming languages.
- ✓ This can be helpful to programmers/developers.

3.5. Music

- ✓ Generative AI models are being used to create new music.
 - Creating music for movies and TV shows.
 - Generating personalized playlists & etc.

4. Artificial Intelligence

- ✓ Artificial Intelligence is the **ability** for a computer to think, learn and do tasks.
 - Problem solving.
 - Understanding Language.
 - Making decisions & etc.
- ✓ AI having **below** topics like,
 - Machine Learning.
 - Deep Learning.
 - Natural Language Processing.
 - Computer Vision.
- ✓ By using above topics we can do,
 - Prediction.
 - Classification.
 - Sentiment analysis & etc.
- ✓ AI can **enable** machines to mimic human.
- ✓ Artificial Intelligence short form is **AI**.

5. AI Model

- ✓ AI model is a **program**.
- ✓ This program **analyses** datasets to find **patterns** and make **predictions**.

6. Generative AI Models

- ✓ In Generative AI, models generate the data.

6.1. Generative Pre-trained Transformer (GPT)

- ✓ **GPT** is Large Language Model.
- ✓ This is developed by OpenAI.
- ✓ It's trained on a massive dataset of text and code.
- ✓ It is Capable of,
 - Generating text.
 - Translating languages.
 - Writing different creative content.
 - Answering your questions.
- ✓ GPT - 4 is the latest version at the time of this writing.

6.2. Llama 2

- ✓ **Llama 2** is Large Language Model.
- ✓ This is developed by Meta.
- ✓ Llama 2 is second version of a natural large language model.

6.3. Claude

- ✓ Claude is Large Language Model.
- ✓ This is developed by startup company called Anthropic.
- ✓ This is like a ChatGPT, it can,
 - Generate text
 - Write code
 - Summarize & etc.

6.4. Gemini

- ✓ Gemini is Generative AI model.
- ✓ This is developed by Google company.
- ✓ It's a Google's new multi-modal model.
- ✓ This can understand,
 - Text.
 - Images.
 - Videos.
 - Audio.
- ✓ It will be available in different sizes (Ultra, Pro, and Nano), each with different capabilities.

6.5. PaLM2

- ✓ PaLM is Large Language Model.
- ✓ The full form of PaLM is Pathway Language Model.
- ✓ It is a multi-modal model
- ✓ This is developed by Google company.
- ✓ This can process,
 - Text.
 - Code.
 - Images.

6.6. DALL-E

- ✓ DALL-E is Visual AI model.
- ✓ This is developed by OpenAI.
- ✓ It can create,
 - Realistic images from text prompts.

6.7. Stable Diffusion

- ✓ Stable Diffusion is an image generation model.
- ✓ This is developed by OpenAI.
- ✓ It can,
 - Generate detailed images.
 - Text descriptions.
 - Inpainting and out painting.
 - Generate image-to-image translations.
- ✓ This model generates these are by prompt as input.

6.8. Midjourney

- ✓ Midjourney is an image generation model.
- ✓ This is developed by startup called Midjourney, Inc.
- ✓ This is like DALL-E and Stable Diffusion.

6.9. CodeWhisperer

- ✓ CodeWhisperer is a **code** generation model.
- ✓ This is developed by AWS.
- ✓ This can generate the,
 - Code in several programming languages. (Python, Java, JavaScript, TypeScript & etc)

6.10. CodeLlama

- ✓ CodeLlama is a large language model.
- ✓ This is built on Llama 2.
- ✓ This model specifically trained on code.
- ✓ It also comes in various sizes and supports multiple popular programming languages.

6.11. Codex

- ✓ Codex is a large language model.
- ✓ This is a **code** generation model.
- ✓ This model specifically trained on code.
- ✓ This can generate the,
 - Code in several programming languages. (Python, C#, Java, JavaScript, SQL, Go, PHP, and Shell).

7. Use cases of Generative AI

- ✓ In Generative AI, models generate the data.

7.1. Content Generation

- ✓ Generative AI helpful to generate the content,
 - Blogs,
 - Reports,
 - e-mails
 - Social media posts.
- ✓ This content helpful to business for marketing.

7.2. Personalized marketing

- ✓ Generative AI can create personalized marketing content,
 - e-mails
 - Landing pages.
 - Social media posts.
- ✓ This content helpful to businesses to reach their target audience more effectively and increase conversion rates.

7.3. Customer service

- ✓ Generative AI can be used to create chatbots that can answer customer questions and resolve issues.
- ✓ This is really great advantage like, free human customer service.

7.4. Risk management

- ✓ Generative AI can identify and predict risks,
 - Fraud.
 - Cyberattacks
 - Supply chain disruptions.
- ✓ This will be helpful to businesses to protect their assets.

7.5. Compliance

- ✓ Generative AI can,
 - Generate compliant documents,
 - Contracts.
 - Reports.
 - Disclosures.
- ✓ This can help businesses to save time and money and reduce the risk of non-compliance.

7.6. Software Development

- ✓ Generative AI can,
 - Generate new code
 - Provide code snippets, or even write simple software.
 - Potentially saving time and reducing errors.
- ✓ In addition, it also helps document code, refactor, generate test cases, and optimize existing code.

7.7. Data Augmentation

- ✓ Generative AI can create synthetic data for Data Science projects if needed.

8. Below domains are using the Generative AI

8.1. Financial

- ✓ Generative AI can help with,
 - Decision-making.
 - Risk model assessment.
 - Development of new financial products and services.
- ✓ Customer operations to enhance services and resolutions for each client based on transactions and history.

8.2. Healthcare

- ✓ Generative AI is used to develop,
 - New drugs and treatments.
 - Design medical devices.
 - Create personalized patient treatment plans.
 - Generate patient documentation on instructions, risks, and drug interactions.

8.3. Manufacturing

- ✓ Generative AI is used to develop,
 - Design new products.
 - Optimize manufacturing processes.
 - Improve quality control.

8.4. Retail and Consumer Packaged Goods

- ✓ Generative AI is used to,
 - Personalize shopping experiences.
 - Recommend products.
 - Manage inventory.
 - Accelerate consumer research.
 - Enhance the supply chain & etc.

8.4. Marketing and Sales

- ✓ Generative AI is helping enhance,
 - Understand real-time customer trends.
 - Personalized outreaches embedded into virtual assistants.
 - Dynamic customer journeys & etc

9. OpenAI's Introduction

- ✓ OpenAI is an artificial intelligence research organization and technology company founded in December 2015.
- ✓ Its mission is to ensure that artificial general intelligence (AGI) benefits all of humanity.
- ✓ OpenAI develops AI models and technologies, including the famous language models like GPT (Generative Pre-trained Transformer).

10. Installation

- ✓ `pip install openai`

Hello World example

11. OpenAI's Hello World Program

- ✓ Let's write and run the basic example by using OpenAI's API.

Kind note

- ✓ To run below program, we should set API key first.

```
Program Name    OpenAI's API Hello World Program
demo1.py

from openai import OpenAI

client = OpenAI()

response = client.completions.create(
    model = "gpt-3.5-turbo-instruct",
    prompt = "How are you."
)

print(response.choices[0].text)
```

Output

I am an AI and do not have the capability to feel emotions.

12. Understanding the Hello World Program

- ✓ **openai** is library, we are importing.
- ✓ **OpenAI** is predefined class, we are importing from openai library.
- ✓ **client** is an object to OpenAI class.
- ✓ **completion** is an object, accessing by using client.
- ✓ **create** is method, accessing by using client.completions.
- ✓ **create** method having two parameters,
 - **model**
 - **prompt**

13. Hugging Face

- ✓ Hugging Face is a leading company in the field of natural language processing (NLP) and machine learning.
- ✓ Founded in 2016, it has become well-known for its contributions to the AI community, particularly through its open-source libraries.

13.1. transformers library

- ✓ pip install transformers
 - It Provides a wide range of pre-trained models for various NLP tasks, such as text generation, translation, summarization, and classification.
 - Supports models like BERT, GPT, and many others.

Program Name Text Classification
demo1.py

```
from transformers import pipeline
```

```
# Load a pre-trained model for text classification  
classifier = pipeline("text-classification")
```

```
# Classify a text  
a = "I love using Hugging Face's transformers library!"  
result = classifier(a)  
print(result)
```

Output

```
[{'label': 'POSITIVE', 'score': 0.9978122711181641}]
```

Program Name Text Generation
demo2.py

```
from transformers import pipeline

# Load a pre-trained model for text generation
generator = pipeline("text-generation")

# Generate text based on a prompt
a = "Once upon a time, in a land far away,"
result = generator(a, max_length = 50)
print(result)
```

Output

```
[{'generated_text': 'Once upon a time, in a land far away, the land of the dead, and the land of a living creature, the human race had become. In them died the dead, and in them dwelt men, who in these times had never'}]
```

Program Name Question Answering
demo3.py

```
from transformers import pipeline

# Load a pre-trained model for question answering
qa = pipeline("question-answering")

# Answer a question based on a context
context = "Hugging Face is an AI company based in New York.
They are known for their work in natural language processing."
question = "Where is Hugging Face based?"

result = qa(question=question, context=context)
print(result)
```

Output

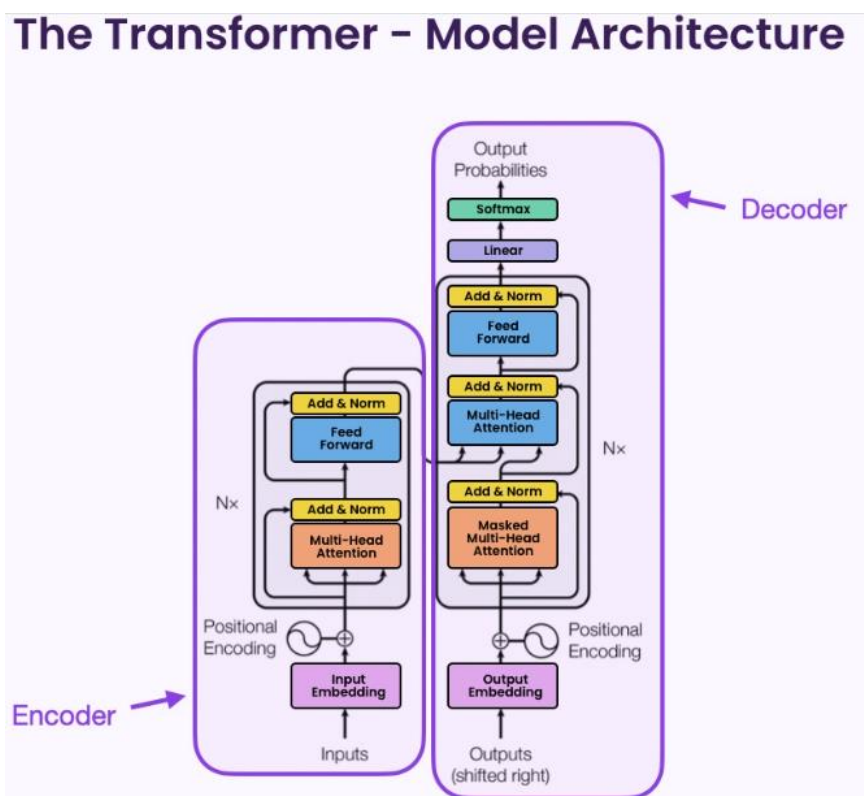
```
{'score': 0.9903117418289185, 'start': 39, 'end': 47, 'answer': 'New
York'}
```

14. Large Language Model (LLM)

- ✓ LLM stands for "Large Language Model".
- ✓ It is a type of AI trained model on text data,
 - To understand human text
 - To generate human-like text
 - Handling tasks like answering questions
 - Summarizing text data
 - Translating the text data
 - Creating new content.

15. Large Language Model Architecture

- ✓ Large Language Models (LLMs) use the Transformer neural network architecture.
- ✓ It is introduced in the 2017 paper "Attention is All You Need" by Vaswani et al.
- ✓ Here we can find effectively processing large-scale data and capturing complex language patterns.



15.1. Input Embedding:

- ✓ **Purpose:** Each word in the input sequence is represented as a high-dimensional vector, capturing **semantic meaning**.

15.2. Positional Encoding:

- ✓ **Purpose:** Adds information about the position of each token. Positional encoding is added to the Transformer model to provide information **about the order of tokens**

15.3. Add & Norm:

- ✓ **Purpose:** Combines the output of a sub-layer with the input and normalizes it. The "Add" operation is a residual connection that helps the model avoid vanishing gradients, while "Norm" refers to Layer Normalization, which stabilizes and **speeds up training**.

15.4. Multi-Head Attention:

- ✓ **Purpose:** Multi-head attention uses several attention mechanisms (heads) in parallel. Each head processes the input differently, allowing the model to capture **various relationships** in the data.

15.5. Feed Forward:

- ✓ **Purpose:** After attention is applied, a feed-forward network processes the data, adding non-linearity and allowing the model to **capture more complex patterns**.

15.6. Encoder Block (Left Side):

- ✓ **Purpose:** Consists of several identical layers (denoted as N_x), each containing Multi-Head Attention, Add & Norm, and Feed Forward sub-layers. Processes the input sequence to **create a context-aware representation** of the input tokens.

15.7. Masked Multi-Head Attention:

- ✓ **Purpose:** In the decoder, this layer masks future positions to ensure that predictions for a token **depend only on previous tokens, not on future ones.**

15.8. Decoder Block (Right Side):

- ✓ **Purpose:** Similar to the encoder, it consists of several identical layers (Nx), but with an additional masked attention mechanism. **Generates** the **output sequence** by attending to both the encoder's output and the previously generated tokens.

15.9. Linear + Softmax:

- ✓ **Purpose:** The linear layer reduces the dimensionality of the decoder's output, and the softmax function converts this into a probability distribution, **predicting** the next token in the sequence.

15.10. Output Probabilities:

- ✓ **Purpose:** The final prediction of the model. The output probabilities are used to determine the next token in the sequence, forming the basis for tasks like translation, summarization, or text generation.