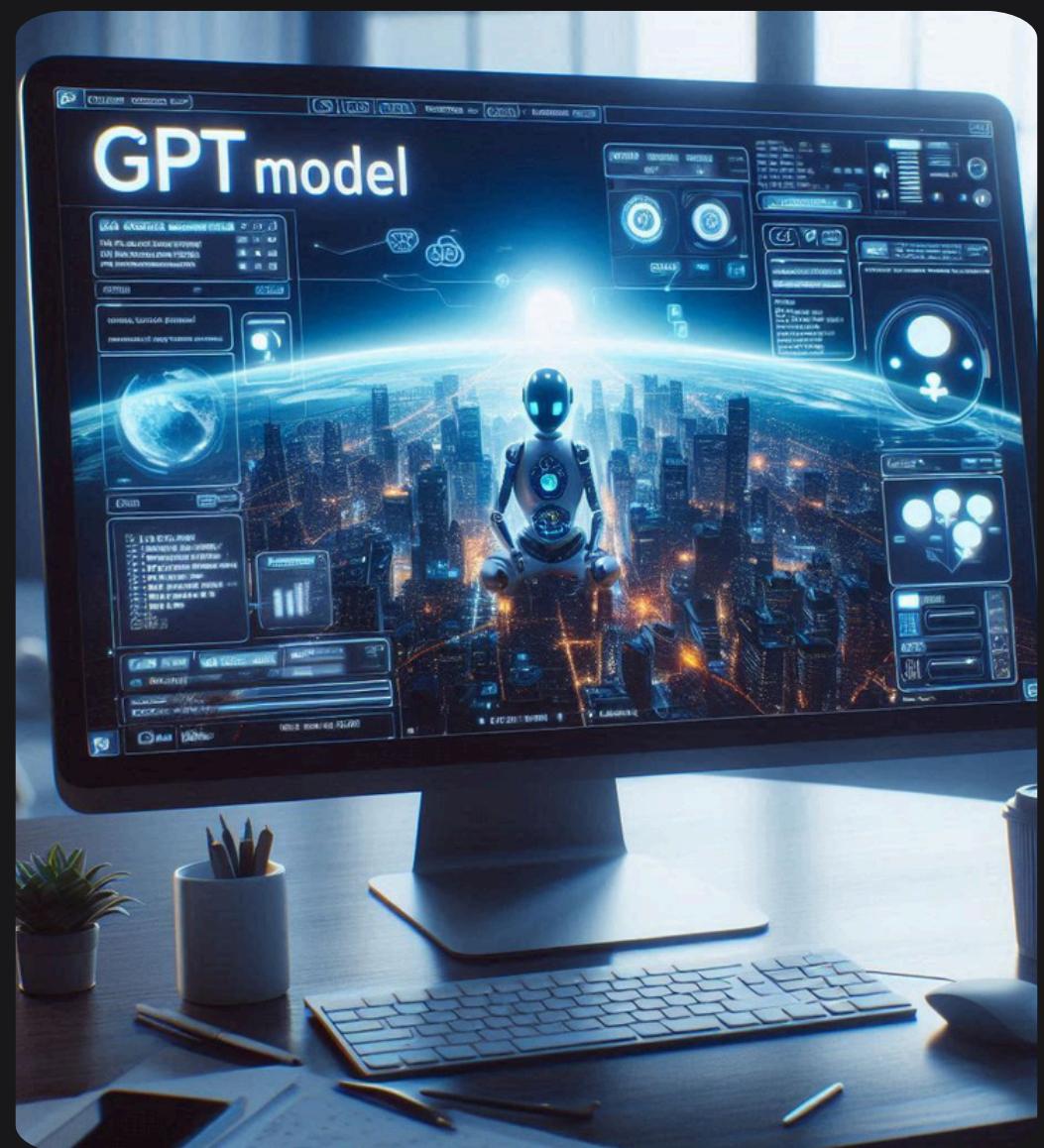


# BUILD A LOCAL-GPT ON YOUR COMPUTER IN 6 STEPS



Run AI Locally on your computer.



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)





# Step 1 - Installing WSL and Ubuntu

## Installing Windows Sub-System for Linux

- Open the Terminal on your PC. Next, install WSL using the following command:  
**wsl --install**
- This process may prompt you to reboot your system. If so, restart your computer and allow the installation to complete automatically after the reboot.
- Once the installation is finished, you will be prompted to set up a UNIX username and password. Follow the prompts to create your account.
- After completing the setup process, wait for the WSL installation to finish. You should eventually see the "Welcome to Ubuntu" message, indicating that the installation was successful.

```
avrsanand@Knight: ~      x + v
Ubuntu is already installed.
Launching Ubuntu...
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: avrsanand
New password:
Retype new password:
passwd: password updated successfully
The operation completed successfully.
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.153.1-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

This message is shown once a day. To disable it please create the
/home/avrsanand/.hushlogin file.
avrsanand@Knight:~$ |
```



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)



# Step 2 - Updating Packages

## Ensuring Latest software

To ensure that your Ubuntu system has the latest software updates, run the following commands:

- ***sudo apt update***: This command updates the package index to reflect the availability of the latest packages in the Ubuntu repository. It's a best practice to keep the package list up-to-date before upgrading or installing new software.
- ***sudo apt upgrade -y***: This command upgrades all installed packages to their latest versions, using the updated package list from the previous step. The *-y* flag is used to automatically answer "yes" to any prompts that may appear during the upgrade process.

```
avrsanand@Knight:~$ sudo apt update  
[sudo] password for avrsanand:
```

```
avrsanand@Knight:~$ sudo apt upgrade -y  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
Calculating upgrade... Done
```



AVRS Anand

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)



# Step 3 - Installing Ollama

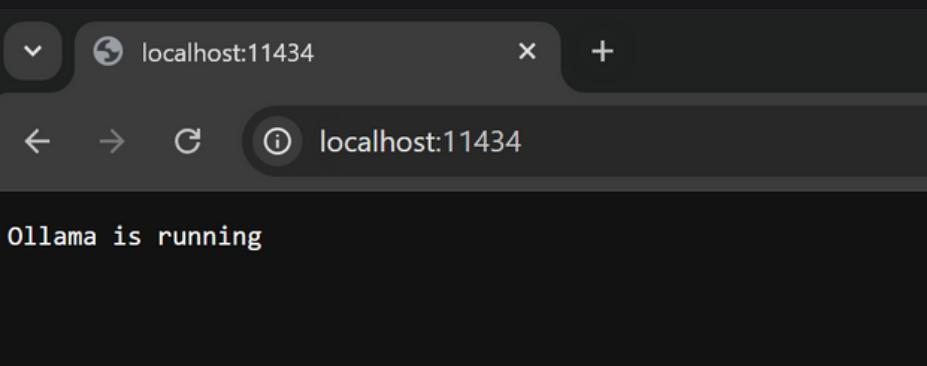
# Installing Ollama

# Follow these steps -

- Visit "Ollama.ai" and select "Linux" as the installation option.
  - Copy the provided code snippet and paste it into your terminal application.
  - Once you've pasted the code, wait for the installation to complete. You'll know it's finished when you see the message "Install Complete".

# Verify Installation

To confirm that Ollama is running successfully, open a web browser and navigate to ***http://localhost:11434***. If Ollama is installed correctly, you should see the Ollama is running.



# AVRS Anand



 [linkedin.com/in/avrsanand/](https://www.linkedin.com/in/avrsanand/)

# Step 4 - Pulling and Running LLaMA model in CLI

## Installing LLaMA3 Model

Use the command - ***ollama pull llama3***. This will download and install the LLaMA-3 model. Once the installation is complete, you should see the output "success".

## Running LLaMA3 in CLI

- To interact with the LLaMA-3 model in a command-line interface (CLI), use the following command:  
***ollama run llama3***
- This will launch the CLI and allow you to chat with the LLaMA-3 model. You can ask it questions or engage in conversation. To quit the conversation type ***/bye***.

```
avrsanand@Knight:~$ ollama pull llama3
pulling manifest
pulling 6a0746a1ec1a... 100% 4.7 GB
pulling 4fa551d4f938... 100% 12 KB
pulling 8ab4849b038c... 100% 254 B
pulling 577073ffcc6c... 100% 110 B
pulling 3f8eb4da87fa... 100% 485 B
verifying sha256 digest
writing manifest
removing any unused layers
success
avrsanand@Knight:~$ |
```

```
avrsanand@Knight: ~
avrsanand@Knight:~$ ollama run llama3
>>> Tell me about LLMs
LLMs (Large Language Models) are a type of artificial intelligence (AI) model that has revolutionized the field of natural language processing (NLP). Here's an overview:
**What is a Large Language Model?**
A Large Language Model (LLM) is a type of neural network designed to process and generate human-like text. It's trained on vast amounts of text data, which allows it to learn patterns, relationships, and structures in language.

**Key Characteristics:**
1. **Scale:** LLMs are massive models, with millions or even billions of parameters. This scale enables them to capture complex linguistic phenomena.
2. **Self-supervised training:** LLMs are trained on large datasets without explicit human supervision. Instead, they learn from the patterns and relationships in the data itself.
3. **Transformer architecture:** LLMs typically use a transformer-based architecture, which is particularly well-suited for processing sequential data like text.
```



AVRS Anand

[linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)

# Step 5 - Installing Docker on Ubuntu for Open WebUI

## Update Package Index for Docker

First, update the package index

- ***sudo apt-get update***

Then install necessary packages for Docker installation

- ***sudo apt-get install ca-certificates curl***
- ***sudo install -m 0755 -d /etc/apt/keyrings.***

## Install Docker's GPG key

Create a new directory and install Docker's official GPG key

- ***sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc***

Set the permissions for the GPG key

- ***sudo chmod a+r /etc/apt/keyrings/docker.asc***

## Add repository to Apt sources

Add Docker's repository to Apt sources

- ***- echo "deb [arch=\$(dpkg --print-architecture) signed  
by=/etc/apt/keyrings/docker.asc]  
https://download.docker.com/linux/ubuntu \$(./etc/os-release && echo  
"\$VERSION\_CODENAME") stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null.***

This allows you to install Docker from the official repository.



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)



# Step 5 - Cont..

## Check added Repository

Update the package index again to reflect the newly added repository using

- ***sudo apt-get update***

## Install Docker Container

Install docker and its command-line interface (CLI), container runtime (containerd.io), buildkit plugin and compose plugin use this command

- ***sudo apt-get install docker-ce  
docker-ce-cli containerd.io  
docker-buildx-plugin docker-  
compose-plugin***

Verify the installation using command

- ***sudo docker ps***

```
avrsanand@Knight:~$ # Add Docker's official GPG key:  
sudo apt-get update  
sudo apt-get install ca-certificates curl  
sudo install -m 0755 -d /etc/apt/keyrings  
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/a  
pt/keyrings/docker.asc  
sudo chmod a+r /etc/apt/keyrings/docker.asc  
  
# Add the repository to Apt sources:  
echo \  
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/do  
cker.asc] https://download.docker.com/linux/ubuntu \  
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \  
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null  
sudo apt-get update  
[sudo] password for avrsanand:  
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [129  
kB]  
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease  
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]  
] Hit:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease  
Fetched 257 kB in 2s (144 kB/s)  
Reading package lists... Done  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
ca-certificates is already the newest version (20230311ubuntu0.22.04.1  
).  
ca-certificates set to manually installed.  
curl is already the newest version (7.81.0-1ubuntu1.17).  
curl set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.  
Get:1 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]  
Get:2 https://download.docker.com/linux/ubuntu jammy/stable amd64 Pack  
ages [37.8 kB]  
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease  
Hit:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease  
Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease  
Hit:6 http://security.ubuntu.com/ubuntu jammy-security InRelease  
Fetched 86.6 kB in 1s (69.9 kB/s)  
Reading package lists... Done  
avrsanand@Knight:~$
```

COLUMN	CONTAINER ID	IMAGE	STATUS	COMMAND	PORTS	NAMES	CREATED	Up
	5a9496f0a61e	ghcr.io/open-webui/open-webui:main	"bash start.sh"	open-webui			About a minute (healthy)	



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)



# Step 6 - Accessing Open



## Using Open WebUI

Now that we have Docker installed, let's access Open WebUI on our local machine. Open a web browser and navigate to ***http://localhost:8080***. You should see the Open WebUI interface.

- Sign-up and Initial Setup: Click "Sign-up" and create an account. The first person to sign up locally will have admin access.
- Accessing the interface: You can view a similar UI to OpenAI's ChatGPT. You need to select the model which you have installed - Llama3

The image shows two screenshots of the Open WebUI interface. On the left, the sign-in page is displayed with fields for Email and Password, and a 'Sign in' button. Below it, a link to 'Don't have an account? [Sign up](#)'. On the right, the main chat interface is shown. It features a sidebar with 'New Chat', 'Workspace', and 'Search' options. A 'Select a model' dropdown is open, showing 'llama3:latest 8.0B' as the selected option with a checkmark. The main area displays a greeting from the AI: 'Hello, AVRS Anand How can I help you today?'. Below this, there are four suggested prompts: 'Give me ideas', 'Explain options trading', 'Help me study', and 'Tell me a fun fact'. At the bottom, there's a message input field with '+ Send a Message' and a note: 'LLMs can make mistakes. Verify important information.' To the left of the screenshots, there is a large, hand-drawn style five-pointed star with radiating lines.

**AVRS Anand**

[linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)

# Exploring Open WebUI

## Features Features Features !!!

You can now use Llama3 locally and there are many features imbedded into Open WebUI.

The screenshot shows the Open WebUI interface for the Llama3:latest model. A red box highlights the sidebar menu which includes 'New Chat', 'Workspace', 'Search', 'Today', and 'Large Language Models'. A white arrow points from the text 'Use multiple chats and search them' to this menu. Another red box highlights the 'Upload Files' button in the message input bar. A white arrow points from the text 'You can upload files and ask model to analyze, answer using the information in those files.' to this button. A third red box highlights the microphone icon in the message input bar. A white arrow points from the text 'Use voice to chat' to this icon. The main content area contains sections on how LLMs generate text, key features, and applications, along with a note about the model's capabilities and a summary statement.

1. A portion of the input text is randomly replaced with a [MASK] token.  
2. The model predicts the original word that was masked, based on the context and patterns learned from the training data.  
3. This process is repeated millions of times to train the model.

**How LLMs generate text:**  
When you ask an LLM to generate text, it uses its trained parameters to:

1. Tokenize the input prompt (e.g., a sentence or phrase).
2. Pass the tokens through the encoder to extract relevant information and context.
3. Use the decoder to generate output text, one token at a time, based on the input and learned patterns.

**Key features:**  
LLMs have several notable characteristics that enable their impressive language generation capabilities:

1. **Contextual understanding**: LLMs can understand complex relationships between tokens in text.
2. **Generative capacity**: They can generate new text that is coherent and grammatically correct.
3. **Flexibility**: LLMs can be fine-tuned for specific tasks or domains, such as conversational dialogue or language translation.

**Applications:**  
LLMs have numerous applications across various industries, including:

1. **Conversational AI**: They power chatbots, virtual assistants, and voice-controlled interfaces.
2. **Content generation**: LLMs can generate news articles, product descriptions, and social media posts.
3. **Language translation**: They facilitate machine translation of languages.
4. **Summarization**: LLMs can summarize long documents or articles.

In summary, LLMs are powerful AI models that learn to understand patterns in language by training on massive datasets. They can generate text based on input prompts and have many applications across various industries.

*You can upload files and ask model to analyze, answer using the information in those files.*

*You can edit your response, like and dislike for improvement, read-aloud along with making the model retry*

*Use voice to chat*



**AVRS Anand**

[linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)



# CREDITS !!

This project was developed through NetworkChuck's youtube video



Follow and Subscribe his Youtube Channel for more amazing videos



[NetworkChuck](#)



**NetworkChuck**  
Welcome to NetworkChuck! I LOVE Information Technology!! My goal is to help as MANY PEOPLE AS POSSIBLE jump into a career in the IT field through...  
[YouTube](#)



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)

# Find this helpful?

-  Like this post
-  Leave a comment
-  Save for later
-  Let's connect!

yes!  
~



**AVRS Anand**

 [linkedin.com/in/avrsanand/](https://linkedin.com/in/avrsanand/)