



VAASAN AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES

Elias Tarsia

INSTALLATION AND MACHINE LEARNING  
OF ARTIFICIAL INTELLIGENCE ON  
RASPBERRY PI PLATFORMS

Technology  
2024

## TIIVISTELMÄ

Tekijä	Elias Tarsia
Opinnäytetyön nimi	Tekoälyn asennus ja koneoppiminen Raspberry Pi -alustoilla
Vuosi	2024
Kieli	Englanti
Sivumäärä	42 + 2 liitettä
Ohjaaja	Jukka Matila

---

Tämä opinnäytetyö keskittyy nopeasti kehittyvään tekoälyalaan, koneoppimiseen ja Raspberry Pi -alustan kelpoisuuteen työkaluna näiden teknologioiden tutkimisessa, käytössä ja kehittämisessä.

Testaus tehtiin kahdella Raspberry Pi -laitteella, malleilla 3 B ja 5, jotka molemmat käyttävät Raspberry Pi -säätöön suosittelemaa uusinta käyttöjärjestelmäversiota. Nämä laitteet valikoituivat testialustoiksi niiden suuren suorituskykyeron ja saatavuuden vuoksi.

Raspberry Pi 5 on erinomainen alusta päästä alkuun tekoälyalalla. Pienempien kielimallien käyttö sujuu sulavasti ja laite tarjoaa erinomaisen alustan koodaamiseen, kehittämiseen ja pienempien kielimallien testaamiseen. Syvä koneoppiminen vaatii kuitenkin liikaa järjestelmäresursseja laitteella suoritettavaksi. Raspberry Pi 3 B on käyttökelpoisin IoT -laitteena, sillä laitteisto ei pysty suorittamaan suurinta osaa tekoälyyn liittyvistä raskaista prosesseista paikallisesti.

VAASAN AMMATTIKORKEAKOULU  
UNIVERSITY OF APPLIED SCIENCES  
Tietotekniikka

## ABSTRACT

Author	Elias Tarsia
Title	Installation and Machine Learning of Artificial Intelligence on Raspberry Pi Platforms
Year	2024
Language	English
Pages	42 + 2 Appendices
Name of Supervisor	Jukka Matila

---

This thesis focuses on the rapidly advancing technology of Artificial Intelligence, machine learning and the viability of the Raspberry Pi platform as a tool to study, run and develop these technologies.

The testing was done on two Raspberry Pi devices, models 3 B and 5, both running the newest version of the operating system recommended by the Raspberry Pi Foundation. These devices were chosen because of their performance differences and availability.

The conclusion is that the Raspberry Pi 5 is a great platform to get started in the artificial intelligence field. Running the smaller large language models is smooth and the device offers a great platform to code, develop and test the smaller models. However, deep machine learning requires too much system resources to be done on the device. The Raspberry Pi 3 B is best used as an IoT device as the hardware is not capable of running most AI related processes.

---

Keywords	Artificial intelligence, machine learning, Raspberry Pi, large language model
----------	---

# CONTENTS

TIIVISTELMÄ

ABSTRACT

1	INTRODUCTION .....	9
2	THEORETICAL BASE.....	10
	2.1 Linux and Raspberry Pi.....	10
	2.2 Raspberry Pi OS.....	10
	2.3 Raspberry Pi .....	11
3	ARTIFICIAL INTELLIGENCE, MACHINE LEARNING AND LARGE LANGUAGE MODELS.....	12
	3.1 Artificial Intelligence .....	12
	3.2 Types of Artificial Intelligence Applications.....	12
	3.3 Machine Learning .....	13
	3.4 Large Language Models .....	14
4	HARDWARE.....	16
	4.1 Component Specifications .....	16
	4.2 Dimensions.....	17
5	DEPLOYMENT OF RASPBERRY PI 3 B AND 5 .....	18
6	INSTALLING AND OPERATING LARGE LANGUAGE MODELS.....	20
7	ARTIFICIAL INTELLIGENCE ON RASPBERRY PI .....	21
	7.1 Raspberry Pi 3 .....	21
	7.2 Extending the Hardware Capabilities of Raspberry Pi 3 .....	21
	7.3 Running an LLM through SSH on a Virtual Machine.....	22
	7.4 Running the LLM's on the Ubuntu Server.....	25
	7.5 Raspberry Pi 5 .....	26
	7.6 Large Language Models on the Raspberry Pi 5.....	27
	7.7 Entry Level Fine Tuning of Large Language Models.....	27
8	HARDWARE PERFORMANCE AND DIFFERENT LLM MODELS COMPARISON .	31

8.1	Model Comparison.....	31
8.2	Hardware Monitoring .....	34
9	MACHINE LEARNING ON RASPBERRY PI.....	37
9.1	Training Large Language Models .....	37
9.2	Possibilities of the Raspberry Pi.....	38
10	CONCLUSIONS .....	39
	REFERENCES .....	41
	APPENDICES .....	43

## LIST OF FIGURES AND TABLES

<b>Graph 1</b> Model performance comparison . <b>Virhe. Kirjanmerkkiä ei ole määritetty.</b>	
<b>Graph 2</b> Model load times .....	34
<b>Table 1</b> Hardware comparison.....	16
<b>Table 2</b> Dimensions .....	17
<b>Table 3</b> VM Hardware .....	26
<b>Table 4</b> Verbose data explanations.....	32
<b>Table 5</b> Model performance comparison .....	32
<b>Table 6</b> Hardware utilization of large language models .....	35
<b>Table 7</b> Yi verbose data .....	43
<b>Table 8</b> Falcon verbose data .....	44
<b>Table 9</b> Notus verbose data .....	45
<b>Table 10</b> Vicuna verbose data.....	47
<b>Table 11</b> Dolphin-Mistral verbose data .....	49
<b>Table 12</b> Solar verbose data.....	51
<b>Table 13</b> Nous-Hermes2 verbose data .....	53
<b>Figure 1.</b> Connecting to the Ubuntu Server on the Raspberry Pi 3 B .....	23
<b>Figure 2.</b> Installing Ollama.....	24
<b>Figure 3.</b> Error, Ollama is not compatible with armv7l processor architecture ...	25
<b>Figure 4.</b> The model file of Phi .....	28
<b>Figure 5.</b> Changed model file of Phi .....	29
<b>Figure 6.</b> Creation of a new modelfile .....	30

## **LIST OF APPENDICES**

**Appendix 1.** Ollama verbose command raw data, including outputs

**Appendix 2.** Glances data of running models

## CONCEPTS AND ABBREVIATIONS

Linux	Operating system
Debian	A complete, open-source Linux based operating system
AI	Artificial intelligence
AGI	Artificial General Intelligence
ML	Machine learning
OS	Operating system
NVMe	Nonvolatile Memory express
BIOS	Basic Input- Output System
IOT	Internet Of Things
SSH	Secure Shell
UI	User Interface
RAM	Random Access Memory
FAT	File Allocation Table
PyPI	Python Package Index

## 1 INTRODUCTION

Raspberry Pi devices are compact, easy to use, versatile and affordable computers about the size of a deck of playing cards. Their versatility comes from the mostly open-source style of the main operating systems they run, which are Linux based. Linux operating systems are often light and offer high customization and freedom of development for their users. This combined with the affordability of the platform creates a lot of possibilities for the Raspberry Pi user.

The goal of this thesis is to research and test installation and machine learning on Raspberry Pi devices. What makes this interesting is the hardware limitations of the small one board computers and the open-ended possibilities of Linux operating systems and the challenges they include. Although many distributions have been striving towards out of the box usage, in Linux, most things do not come ready made as they do in Windows or Mac operating systems.

The generalization of Artificial Intelligence was greatly fueled by the late November 2022 release of ChatGPT by OpenAI. ChatGPT is a free, online chatbot which outperforms previous virtual assistants such as Apple's Siri or Microsoft's Cortana on every metric, greatly. The free availability and online interface which is accessible on any device and from anywhere has exploded both the corporate worlds and the end users' interest towards artificial intelligence. The development speed of AI has been accelerating greatly since. Many competitors such as Google's Gemini and others have been released since, and smaller groups of developers are releasing custom large language models on Huggingface daily.

## **2 THEORETICAL BASE**

### **2.1 Linux and Raspberry Pi**

Linux operating systems are Unix based OS's that are built on the Linux core or kernel which is a freely available open-source platform to build and modify operating systems on. The Linux kernel was first developed by Linux Torvalds in 1991 on top of the previously existing Unix operating system. [1.]

Linux itself is not an operating system that can be installed and used out of the box as it is shared in different distributions modified to their specific use cases and scenarios. Distributions can vary from non-graphical operating systems that are used fully in the terminal to graphical operating systems that may or may not mimic Windows operating systems. Most popular Linux distributions in everyday desktop use are graphical operating systems such as Linux Mint, Ubuntu, Arch and Debian. For example, Raspberry pi OS (Previously Raspbian) is the Raspberry Pi Foundation's own modified operating system built on Debian. Other than that, Linux is still the leading operating system in server use. An example of a Linux OS for server implementations is SUSE Enterprise Server operating system.

The Linux kernel and many operating systems built on it are fully open source but as the operating systems have gained popularity and the groups providing them have grown into profitable companies, this seems to be changing. For example, Ubuntu was previously open source, but nowadays some features are proprietary.

### **2.2 Raspberry Pi OS**

Previously, the recommended operating system for Raspberry Pi platforms was an OS called Raspbian which is still available and being developed today. The recommended OS has since changed as the Raspberry Pi Foundation has released their own operating system, called simply Raspberry Pi OS. The operating system is installed on an SD card. When the SD card is placed in the Raspberry Pi's SD card

slot, it automatically boots from the SD card and the operating system can be further configured and installed. In the case of non-graphical operating systems such as the Raspberry Pi OS Lite no configuration is needed and the device boots straight into terminal. [2.]

### **2.3 Raspberry Pi**

Raspberry Pi devices are compact and inexpensive single-board computers that offer high customizability, and the hardware data is widely available. The availability of hardware details and the modifiability of Linux creates a platform with an extremely wide range of possibilities. The Raspberry Pi Foundation was started in the United Kingdom in 2008 and has been developing the Raspberry Pi devices since. The original idea was to make inexpensive computers to teach computer science in schools. As the project gained popularity the scope gained width and nowadays the devices are widely used in industry and by hobbyists in different kinds of projects. [3.]

Five generations of Raspberry Pis have been released and the newest is aptly named Raspberry Pi 5. This thesis uses a Raspberry Pi 5 and a Raspberry Pi 3 model B to research the compatibility of AI and Machine Learning on the platforms. Using two different devices gives a good comparison of the hardware capabilities of the newer Raspberry Pi 5 and the older version 3 B. The hardware will be discussed in detail later in this thesis. The foundation has also branched out from making just fully fletched one-board computers to making simpler devices like the Raspberry Pi Zero and Pico. On top of that they make “hats” which are extensions that can be plugged into the devices IO pins and simple peripherals.

Raspberry 5 has introduced a new feature, a power button which allows the device to be rebooted without taking out the power cable and putting it back in or installing the user’s own power button. Another new addition is the PCIe 2.0 port which is an interface for fast peripherals like NVME storage. Other features of the Raspberry Pi 5 mostly improve upon previously existing features.

## **3 ARTIFICIAL INTELLIGENCE, MACHINE LEARNING AND LARGE LANGUAGE MODELS**

### **3.1 Artificial Intelligence**

Artificial intelligence, or in short AI, is the simulation of human intelligence by computer systems. AI systems work by analyzing large amounts of data, its correlations and patterns and then creating output based on mathematical predictions based on said data. In example, small scale AI algorithms are used by GPS systems, for example Google Maps, in calculating the fastest and most efficient routes for the user. Other examples include virtual assistants such as Apple's Siri and recently released ChatGPT and Gemini, which are large language models designed to give answers to users prompts, translate languages, and generate text. All the previously mentioned examples are part of the category narrow or weak Ai. This basically means that the AI must be built from the ground up and trained manually. Strong or general artificial intelligence or AGI is still theoretical and has not been reached yet. AGI would mean that the artificial intelligence would be able to use previously learned data and skills to adapt to completely new scenarios without human intervention. Basically, this would mean that the AI could train itself. [4.]

### **3.2 Types of Artificial Intelligence Applications**

Beyond the previously mentioned AI is used in a variety of applications. Huge advancements have been made in the areas of computer vision, natural language processing and generative adversarial networks.

Computer vision is a field of AI where the algorithm is trained to understand and interpret visual information from the world through cameras, videos, or images. A computer vision model can be fed a picture, and with the proper language model attached it can give a detailed description of what is in the picture. A good example of this is an AI model from the Ollama library called Llava. Computer vision is also

used in facial recognition in most up-to-date smart devices and is being implemented in self-driving cars or more commonly, in lane detection in newer car models.

Natural language processing is the AI's ability to interpret, generate and manipulate human language, meaning that it can analyze the meaning of text and speech, including context, sentiment, and intent behind the words. This is used in virtual assistants, all large language models, and AI chatbots. It recognizes the patterns of a language and with a large amount of data can generate text and translate to other languages the AI is trained on.

These are just a few examples of where AI is applied and used. The field is constantly evolving, and new applications are emerging across industries. In the future AI will affect most, if not all industries.

### **3.3 Machine Learning**

Machine learning or ML is an integral part of AI. It means the ability of the computer to independently develop the algorithms and statistical models needed for the functionality of artificial intelligence. Machine learning algorithms require a substantial amount of training data to learn patterns, relationships, and correlations. The training data is then analyzed, and a testing or validation set is used to evaluate the performance and accuracy. Machine learning can be categorized by the type of training it is given. Examples of types of learning would be Supervised, unsupervised and reinforcement learning. In supervised learning, the correct output is given in the training process, in unsupervised learning, the AI is given the data raw and is forced to find the correct answer through its training parameters. In reinforced learning, the algorithm learns by interacting with an environment and by creating the data itself from the gained feedback. ML learning algorithms include linear regression, decision trees, support vector machines and neural networks. Deep learning, a subset of machine learning has shown great advancement in the line of image and speech generation. [5.]

### 3.4 Large Language Models

Large language models are a specific type of AI that are designed to process and understand large amounts of data. LLMs are trained on datasets of large amounts of text data. The core of the LLM consists of the neural network and the transformer that takes care of transforming the information of the datasets to readable, accurate outputs. Each word is first transferred to numerical form, allowing the network to understand it mathematically. The neural network then analyzes the relationship between the words, considering the order, context and how they contribute to the overall meaning. Over time, the network learns the patterns and statistical relationships within the vast amount of data. This allows it to understand the language, including grammar, syntax, and even different writing styles, if given enough data. [6.]

LLMs range from very small models, fine-tuned to function, for example, on a website as an organization's first point of contact chatbot to text generators that can write whole books from prompts in short periods of time. LLMs are essentially algorithms that are trained on large datasets of different types of text. Depending on the specific large language model use cases, this can range from fantasy books to medical or law texts to code. Most general use large language models are trained with a bit of all previously mentioned. A good example of an AI model that is trained with multiple sets of data would be the LLM called Bloom, released by BigScience. According to the description on hugging face it is an autoregressive model trained to continue text from a prompt on vast amounts of text data. It is able to generate coherent text in 46 languages and 13 programming languages. The description states that the output is "hardly distinguishable from text written by humans". [7.]

Most LLMs that are runnable consumer grade hardware come in 7 and 13 billion parameter models, the size of smaller and larger models vary. The smallest are 2 to 3 billion parameter models and the largest model on Huggingface is 176 B parameters. Up to seven billion parameters can be run on systems with only 8Gb of

RAM and the larger 13B models require 32Gb of RAM to run smoothly. Most of the examples used in this thesis are on the smaller side as the Raspberry Pi 5 is only able to run up to 7 B parameter models smoothly, as depicted later in the thesis. With smaller models, great output speeds are achievable on hardware equal to the Raspberry Pi 5, but the smaller models are prone to “hallucinate” answers to prompts and so give nonfactual information.

## 4 HARDWARE

### 4.1 Component Specifications

The hardware specifications of Raspberry Pi model 5 and 3 B are compared in Table 1. These two devices are compared as they are the devices used as test platforms in this thesis. It is immediately apparent that the Raspberry Pi 5 is a considerable improvement on Raspberry Pi 3 B, doubling, or multiplying efficiency in most fields. [8.]

**Table 1.** Hardware comparison

Model	Raspberry Pi 5	Raspberry Pi 3 B V1.2
CPU	2.4GHz quad-core 64-bit Arm Cortex-A76 CPU	Quad Core 1.2GHz Broadcom BCM2837 64bit
GPU	VideoCore VII GPU, supporting OpenGL ES 3.1, Vulkan 1.2	Dual Core VideoCore IV
Memory (RAM)	8 GB RAM	1 GB RAM
Ethernet	Gigabit Ethernet	100 Base Ethernet
Wi-Fi	Dual-band 802.11ac Wi-Fi	BCM43438 wireless LAN
Bluetooth	Bluetooth 5.0 / Bluetooth Low Energy (BLE)	Bluetooth Low Energy (BLE)
USB	2 x USB 2.0  2x USB 3.0	4 x USB 2.0

## 4.2 Dimensions

The dimensions of the two devices are as given in Table 2.

**Table 2.** Dimensions

Raspberry Pi 5	Raspberry Pi 3 B V1.2
86mm * 56mm * 16	85mm * 56mm * 16mm

## 5 DEPLOYMENT OF RASPBERRY PI 3 B AND 5

The devices Raspberry Pi 3 B and 5 were used as the platforms of this thesis as they have vastly different system resources and the writer of this thesis already has a 3 B device, the model 5 device was purchased specifically for this thesis. Installing a preferred operating system on a Raspberry Pi device is different from installing a Linux or Windows operating system on a PC desktop device. On Raspberry Pi devices there is no need to worry about boot load order or setting up BIOS settings, for example. Only an SD card is needed on which to install the Operating system. The SD card needs to be converted to FAT32 storage, which is a filesystem developed by Microsoft. The filesystem is no longer used in large storage SSDs, or such that normally personal computers use. FAT32 is still used in small USB Stick storages and such. Raspberry Pi OS is designed to boot from FAT32 storage without further configuration.

The easiest way to get an operating system ready for installation is to download the official Raspberry Pi imaging software, called Raspberry Pi Imager, from the official website and to flash and operating system image on the SD card. The software is extremely simple to use, to such extent that anyone with little interest could use it. The software starts up into the main windows where the user can choose the device they are using, the OS they want and the storage they want to flash the OS image on to. The imager software will automatically format the drive and convert it into a FAT. Care should be taken if the user's PC has multiple storages not to choose any of those instead of the SD card, as they might format their storage drive and lose all files on it. The Imager downloads the OS through internet connection and installs it on the SD. The speed of the installation is dependent on the user's internet download speed and the writing speed of the SD card as well the computers processing power where the Imager is running from.

When the device, desired OS and storage are chosen, the software prompts the user with a screen asking the user to choose either a clean install or to edit settings. If editing the settings is chosen, an OS customization window pops up, where the user can give information, such as username, hostname, Wireless LAN credentials and locale settings. With these set, the device is ready to be plugged into a screen and peripherals and for use. If these are not done, the user will be prompted to give the information when the first plugging in the raspberry Pi device.

After the clean installation and plugging the Raspberry Pi 5 in, the device requires the location, user information and prompts to connect to Wi-Fi and to look for device updates. Wi-Fi and updates can be skipped and looked into later. The only difference in setting up the OS in the Imager and on the first start of the device is that setting up a password for the user cannot be skipped when the setup is done on the device itself. After the setup information has been given the device restarts straight onto the desktop. The process from the first plug-in to desktop takes about two minutes without connecting to Wi-Fi and updating the operating system.

## 6 INSTALLING AND OPERATING LARGE LANGUAGE MODELS

Large Language Models or LLM's can be ran locally on a device of choice. The easiest available method to do so right now is Ollama. Ollama is a framework which controls a set of compatible LLM's. Ollama runs in the Linux or Mac OS terminal, and it is going to be released on Windows, but no launch date is announced. Ollama is installed with one terminal command and the large language models are downloaded by either the pull or run command. The run command performs the functionality of the pull command if the LLM is not already downloaded to the system, after which the LLM starts running. After this, with the command for example "ollama run mistral" an LLM is run locally and is ready, after a seemingly non-existent load time even on the Raspberry Pi 5, to answer the users prompts. [9.]

Models available right now for Ollama include Llama, Llama2, Mistral, Llava, Mixtral and on 25<sup>th</sup> of January 2024, fifty-seven others. This library contains general chat LLM's like Mistral and Llama, uncensored models like Dolphin-Mixtral and models trained for coding like Deepseek-coder. These are large language models from different producers and have differences in speed and answer quality. Some models such as wizard-vicuna-uncensored are, as stated in the name, uncensored. Most of the large companies that produce chat based LLM AI's censor the AI in training so it will not give users answers for possible harmful questions. An uncensored model does not have these restrictions. Wizard-vicuna-uncensored is available in 7B, 13B and 30B models and is based on Llama and Llama 2. There are sets of small 7B parameter LLM's like Mistral and larger sets like orca-mini which ranges from 3B to 70B parameters. Orca-Mini is based on Llama 2.

## **7 ARTIFICIAL INTELLIGENCE ON RASPBERRY PI**

### **7.1 Raspberry Pi 3**

The hardware of Raspberry Pi 3 B devices is not enough to run any ready-made large language models, as even the smallest LLM's are designed to require 8 gigabytes of RAM to run smoothly. There is no variety of RAM amounts on the 3<sup>rd</sup> platform, whereas the Raspberry Pi 5 comes in 4Gb and 8Gb models. The Raspberry Pi 3 B can handle light loads in running software, but for example web browsing is quite slow as modern browsers eat up a lot of the device's RAM and system resources. For example, watching a YouTube video on the device is already quite taxing for the hardware.

As the hardware is not capable of running these models, Raspberry Pi 3 B is best used as an IoT device, meaning that it is used as an interface to connect to a server or a service where the LLM is run. A good example of a service where this could be achieved is Google Cloud or Microsoft Azure, where the user is able to create virtual machines with the computing power necessary for the task they have planned. With these tools, anyone could, for example, start their own chatbot service such as ChatGPT and run it from the cloud on the hardware of the company they choose to buy from. In this thesis, the virtual machines needed to run an LLM were created on the Raspberry Pi 3 B.

### **7.2 Extending the Hardware Capabilities of Raspberry Pi 3**

Although Raspberry Pi 3 has low processing power, it can be extended with external processing units like Intel's Movidius Neural Compute Stick, XaLogics K210 AI Accelerator, which is specifically designed for AI applications, and Google's Coral USB accelerator, also designed for AI and ML applications. The device can also be overclocked to achieve better processor performance, but this requires heat control by fans or at least a heatsink to keep the device's components safe. The K210 model is a HAT designed specifically for Raspberry devices and fits into the GPIO

pins directly while the Coral is connected to almost any device via USB. These devices work by offloading task like a machine learning inference from the Pi's CPU to their dedicated hardware, leading to notable performance improvements. This enables Raspberry Pi to handle more complex AI applications, particularly in areas such as image recognition and object detection, expanding its potential for various research and development projects within the realm of artificial intelligence and machine learning. Especially the K210 unit is designed to execute object detection tasks on the board itself.

### **7.3 Running an LLM through SSH on a Virtual Machine**

The Following test used Raspberry Pi 3 as an IoT device. The device was used to control an Ubuntu Server via SSH. Firstly, the virtual machine was created and started, the Ubuntu Server allows the choice to install SSH server in options of the

OS installation. After the server was up and running, a connection was established from the Raspberry Pi 3. This is shown in Figure 2.

```
elias@raspberrypi:~ $ ssh elias@[redacted]
The authenticity of host '[redacted]' can't be established.
ECDSA key fingerprint is SHA256:[redacted].
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[redacted]' (ECDSA) to the list of known hosts.
elias@[redacted]'s password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-92-generic x86_64)

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

System information as of Tue Jan 30 06:02:37 PM UTC 2024

System load:          0.0107421875
Usage of /:           18.1% of 38.09GB
Memory usage:        1%
Swap usage:          0%
Processes:           107
Users logged in:     1
IPv4 address for enp0s3: [redacted]
IPv6 address for enp0s3: [redacted]
IPv6 address for enp0s3: [redacted]

Expanded Security Maintenance for Applications is not enabled.

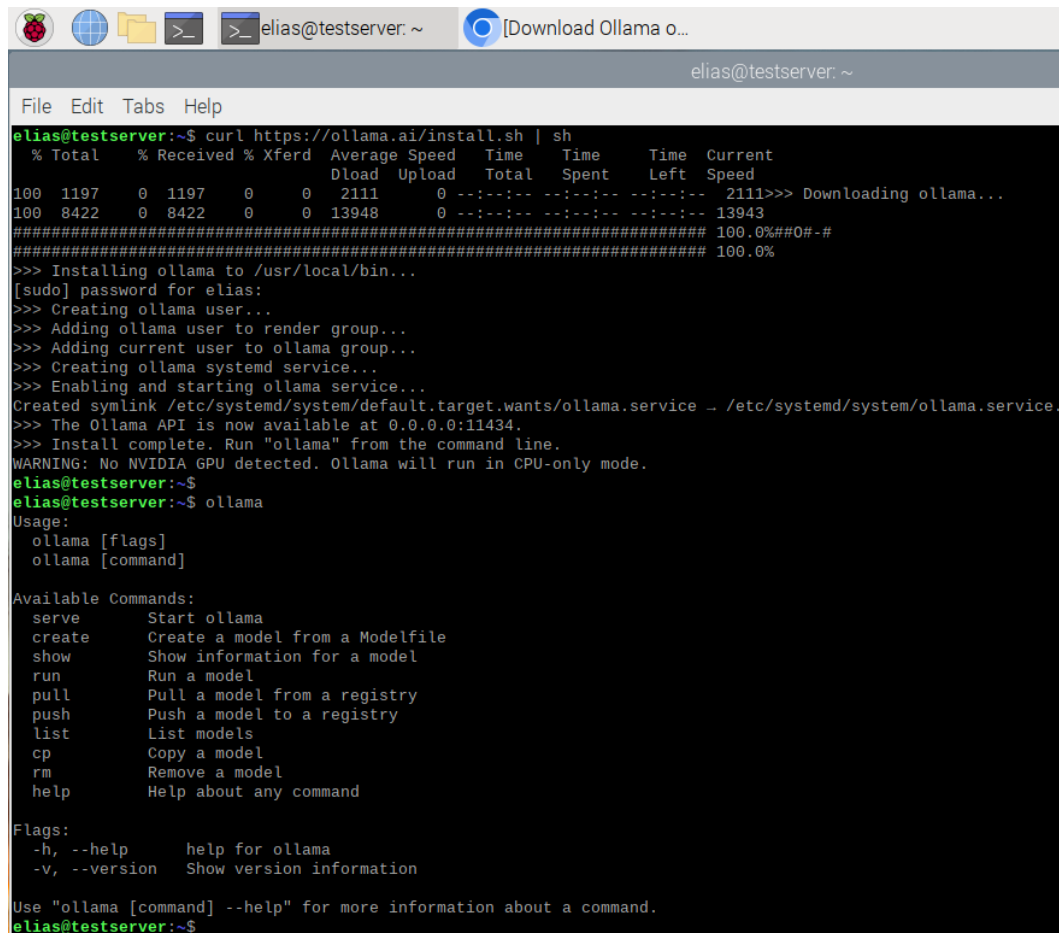
55 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Jan 30 17:59:00 2024
elias@testserver:~$
```

**Figure 1.** Connecting to the Ubuntu Server on the Raspberry Pi 3 B

The connection was established and Ollama and the LLM of choice could be installed, as shown in Figure 3.



```

elias@testserver:~$ curl https://ollama.ai/install.sh | sh
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 1197    0 1197    0    0    2111    0 --:--:-- --:--:-- --:--:-- 2111>>> Downloading ollama...
100 8422    0 8422    0    0 13948    0 --:--:-- --:--:-- --:--:-- 13943
##### 100.0%##0#-#
##### 100.0%
>>> Installing ollama to /usr/local/bin...
[sudo] password for elias:
>>> Creating ollama user...
>>> Adding ollama user to render group...
>>> Adding current user to ollama group...
>>> Creating ollama systemd service...
>>> Enabling and starting ollama service...
Created symlink /etc/systemd/system/default.target.wants/ollama.service → /etc/systemd/system/ollama.service.
>>> The Ollama API is now available at 0.0.0.0:11434.
>>> Install complete. Run "ollama" from the command line.
WARNING: No NVIDIA GPU detected. Ollama will run in CPU-only mode.
elias@testserver:~$ ollama
Usage:
  ollama [flags]
  ollama [command]

Available Commands:
  serve      Start ollama
  create     Create a model from a Modelfile
  show       Show information for a model
  run        Run a model
  pull       Pull a model from a registry
  push       Push a model to a registry
  list       List models
  cp         Copy a model
  rm         Remove a model
  help       Help about any command

Flags:
  -h, --help      help for ollama
  -v, --version    Show version information

Use "ollama [command] --help" for more information about a command.
elias@testserver:~$

```

**Figure 2.** Installing Ollama

Ollama is extremely user friendly to install and use as the installation is achieved by one terminal command, shown in Figure 3. The command is on the Ollama website and can be copied and pasted straight to the terminal. Two problems might occur here, the first being that the curl command is not installed. The curl library can be installed with “sudo apt install curl”. The second problem is system requirements. The error when trying to install Ollama straight onto the Raspberry Pi 3 is depicted in Figure 4. The error is received because the Ollama software is not compatible with the armv7l processor architecture.

```

elias@raspberrypi: ~
File Edit Tabs Help
elias@raspberrypi:~$ curl https://ollama.ai/install.sh | sh
% Total % Received % Xferd Average Speed Time Time Time Current
         Dload Upload Total Spent Left Speed
  0     0     0     0     0     0     0     0  --:--:-- --:--:-- --:--:-- 0ERROR Unsupported architecture: armv7l
100 8422     0 8422     0     0 12760     0  --:--:-- --:--:-- --:--:-- 12741
(23) Failed writing body
elias@raspberrypi:~$

```

**Figure 3.** Error, Ollama is not compatible with armv7l processor architecture

#### 7.4 Running the LLM's on the Ubuntu Server

The hardware of the virtual machine is depicted in Table 3. Having 14Gb of RAM and 6 processor cores the 7B parameter LLM's run well, and their use is absolutely viable for normal, hobbyist use, which does not require extremely fast response times. Faster response times could be achieved by having a more up- to- date PC running the server. The most critical component when running these LLM's is RAM. The SSH connection uses only 2% of the Raspberry Pi 3 Bs processing power and 18 – 19% of the devices RAM. The processing power requirement is surprisingly minimal as even running Firefox with a few tabs open eats up almost all the devices 1Gb of RAM. The server is controlled smoothly, and all the actions are fast. The user experience is not lacking in any way. It has to be stated that the server was connected over Wi-Fi and the devices existed in the same network, so the latency was minimal in this situation.

On the virtual machine hardware mentioned in Table 3 and the Ubuntu server the Mistral LLM reached 122 word per minute, which is greatly faster than the average typing speed. On the other hand, WizardCoder 13B LLM gave a printing speed of 7,8 words per minute, which is too slow for pretty much any type of use.

These tests were done by asking the Mistral LLM to print out a five-hundred-word text about anything and stopping the printing after exactly one minute. The WizardCoder was run for five minutes with a prompt "Can you give me a Hello World program in Python?", and the output was then divided by five. The output did not finish.

**Table 3.** VM Hardware

Ubuntu Server VM hardware specifications	
RAM	14044 MB
Processor (AMD Ryzen 5 3600 6 – Core)	6 cores
GPU	-
Execution cap (processor)	90%

### 7.5 Raspberry Pi 5

As Raspberry Pi 5 has well over double the processing power and eight times the RAM (The Raspberry Pi 5 also comes in a 4 Gb version, 8Gb version is used here) as its predecessor 3 B, the device is well able to run the smaller large language models locally. The output speed is slow compared to modern desktop hardware and the device heats up greatly, even having the standard fan case. Still, the output speed is not slow enough to make the user experience unusable. Smaller than 7B models work fast, but they give less accurate answers and have a narrower scope of use.

The case comes with a 12mm x 17mm x 14mm aluminum heat sink that is installed over the processor with adhesive and a fan that is connected to the boards own fan connector. The fan requires 5V DC and spins 8000 RPM +/- 15% giving adequate airflow for the device. The fan speeds up to 100% when temperature of 75° Celsius is reached. Raspberry Pi 5 starts throttling its processor when temperatures of 85° or greater are reached. These 80 to 90° temperatures were easily reached when the device was sitting the right way up on the desk. Turning the

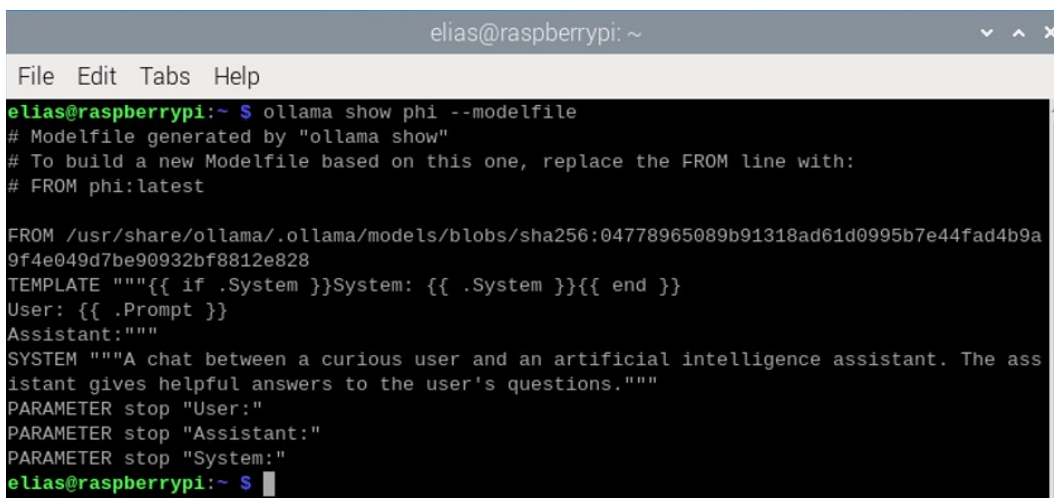
device 90 degrees to sit on its side or propping it up, drops these temperatures to the comfortable 70 – 80° Celsius range while running the LLM's as the air intake is in the bottom of the device. After the output is finished the temperature drops back to 50 – 60° range in less than 3 seconds.

## **7.6 Large Language Models on the Raspberry Pi 5**

As previously mentioned, running the smaller large language models on Raspberry Pi 5 is completely viable. Mistral 7B model takes a while to start before giving replies to prompts but does so with an average typing speed of 40 to 60 words per minute, depending on the length and complexity of the prompt. While running the LLM uses 99.5 to 100% of the devices processing power. A smaller model like Phi, which is a 2.7B parameter LLM developed by Microsoft Research. In their own words the language model “demonstrates outstanding reasoning and language understanding capabilities” [10]. The small LLM runs very well on Raspberry Pi 5, it prints answers fast and takes very little time to load, again depending on the complexity and length of the prompt and the answer. Even the smaller models always take 90 to 100% of the devices processing power. The hardware usage, generation times and differences between models will be looked into in depth in Chapter 10.

## **7.7 Entry Level Fine Tuning of Large Language Models**

The Phi model is easily guided to meet the needs and likings by modifying the model file of the large language model.



```

elias@raspberrypi: ~
File Edit Tabs Help
elias@raspberrypi:~ S ollama show phi --modelfile
# Modelfile generated by "ollama show"
# To build a new Modelfile based on this one, replace the FROM line with:
# FROM phi:latest

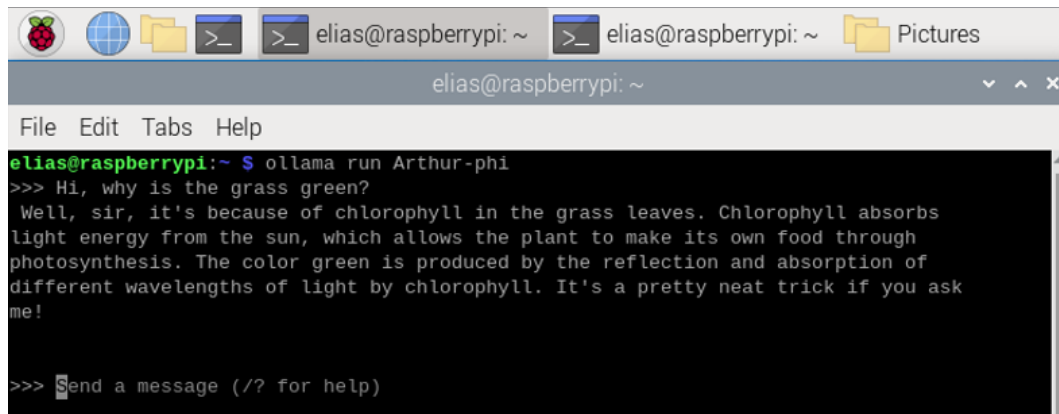
FROM /usr/share/ollama/.ollama/models/blobs/sha256:04778965089b91318ad61d0995b7e44fad4b9a
9f4e049d7be90932bf8812e828
TEMPLATE """{{ if .System }}System: {{ .System }}{{ end }}
User: {{ .Prompt }}
Assistant: """"
SYSTEM """"A chat between a curious user and an artificial intelligence assistant. The ass
istant gives helpful answers to the user's questions.""""
PARAMETER stop "User:"
PARAMETER stop "Assistant:"
PARAMETER stop "System:"
elias@raspberrypi:~ S

```

**Figure 4.** The model file of Phi

The model file is show in Figure 5. The model file allows surface level controlling of the LLM's training data. Changing the system prompts contents allows to modify how the LLM acts. Leaving the system prompt as it is, the model will give answers to prompts in the typical, robotic, formal, and informational way.

Changing the system prompt of the model file, for example, lets us set a character for LLM. This would work much better if the LLM had more information on the character in question. In this test, the system prompt was modified to mimic the way Arthur Morgan from the video game Red Dead Redemption 2 would speak. The LLM has very limited information on who the character is but is able to guess that the character should talk like an American with a southern accent. The light weight 2.7B parameter LLM after modification changes the way it acts, mainly in the way it greets the user as seen in the first test in Picture 6. The LLM now starts its answers in an old fashioned American English way.

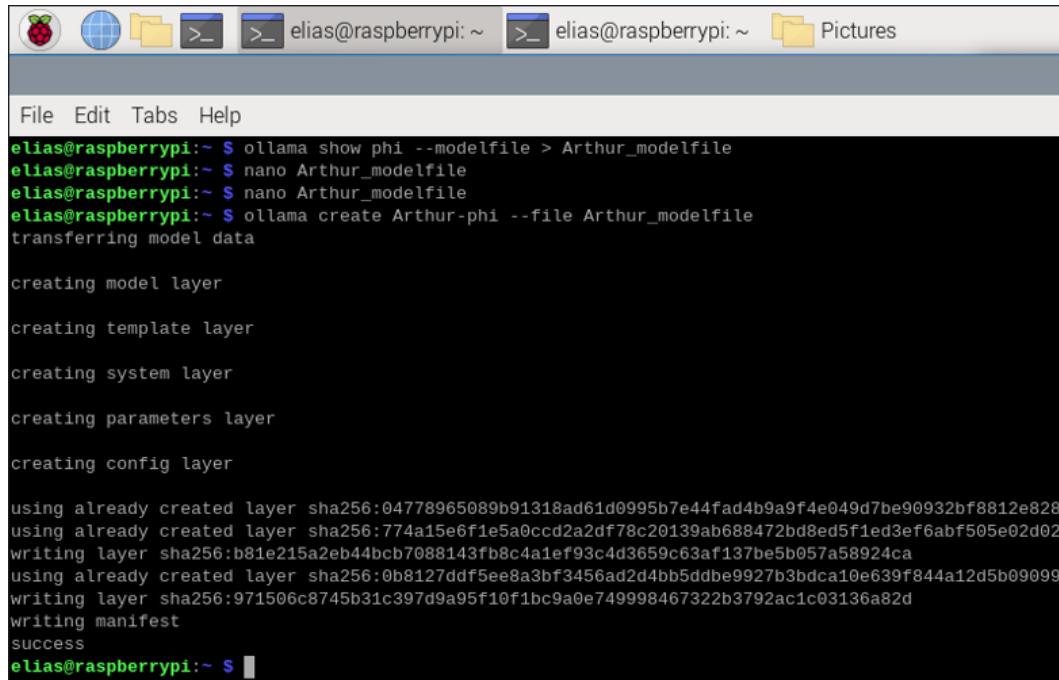


```
elias@raspberrypi: ~  
File Edit Tabs Help  
elias@raspberrypi:~ $ ollama run Arthur-phi  
>>> Hi, why is the grass green?  
Well, sir, it's because of chlorophyll in the grass leaves. Chlorophyll absorbs  
light energy from the sun, which allows the plant to make its own food through  
photosynthesis. The color green is produced by the reflection and absorption of  
different wavelengths of light by chlorophyll. It's a pretty neat trick if you ask  
me!  
>>>  Send a message (/? for help)
```

**Figure 5.** Changed model file of Phi

As we see from the command “ollama run Arthur-phi”, we are now running a new large language model, which was created with the modified system prompt. The Phi model file was first copied into its own file, “Arthur\_modelfile”, and modified with the Nano text editor, after which the LLM was created again with the command “ollama create Arthur-phi --file Arthur\_modelfile”. The whole process with

the terminal commands is depicted in Figure 7.



```

elias@raspberrypi: ~
File Edit Tabs Help
elias@raspberrypi:~ $ ollama show phi --modelfile > Arthur_modelfile
elias@raspberrypi:~ $ nano Arthur_modelfile
elias@raspberrypi:~ $ nano Arthur_modelfile
elias@raspberrypi:~ $ ollama create Arthur-phi --file Arthur_modelfile
transferring model data

creating model layer
creating template layer
creating system layer
creating parameters layer
creating config layer

using already created layer sha256:04778965089b91318ad61d0995b7e44fad4b9a9f4e049d7be90932bf8812e828
using already created layer sha256:774a15e6f1e5a0ccd2a2df78c20139ab688472bd8ed5f1ed3ef6abf505e02d02
writing layer sha256:b81e215a2eb44bcb7088143fb8c4a1ef93c4d3659c63af137be5b057a58924ca
using already created layer sha256:0b8127ddf5ee8a3bf3456ad2d4bb5ddbe9927b3bdca10e639f844a12d5b09099
writing layer sha256:971506c8745b31c397d9a95f10f1bc9a0e749998467322b3792ac1c03136a82d
writing manifest
success
elias@raspberrypi:~ $

```

**Figure 6.** Creation of a new modelfile

As we see, the process of modifying the LLMs with Ollama is extremely user friendly though this should not be confused with machine learning, yet. At most, this is surface level fine tuning of the LLM. Fine tuning means adding information to the LLM's with prompts such as the system prompt that was created in the Arthur-phi test.

## **8 HARDWARE PERFORMANCE AND DIFFERENT LLM MODELS COMPARISON**

Glances was used to monitor the hardware while running the LLM's locally on the Raspberry Pi 5. Glances is an open-source tool which presents a great amount of hardware information in the terminal of the device, or through the built-in Web UI. In this case, the software was used in the terminal. Glances is written in Python, making it almost completely cross platform [11.]. First, the different models will be presented that were used to run these tests.

### **8.1 Model Comparison**

Ollama was built with a feature named "verbose" that prints out data of the prompts and the response which was used to compare different models. Explanations of the data the verbose feature gives the user in Table 4. Every model was tested with the same prompt "Write a 100 word paragraph about AI". Over 7 billion parameter models were also tested, but the tests were stopped after half an hour after just a few words had been printed. The larger models, which comfortably run on 32 gigabytes of RAM, cannot utilize the hardware of the Raspberry Pi 5 so they do not run properly. The tested models were Orca2 13 B, and Qwen 14B. Orca2 started up, but the output speed was about 1 word or character in 10 minutes, this included punctuation marks. Qwen was started and left to load for exactly 30 minutes, and it did not run. A shorter prompt was used with Orca2 which was "Write a 10 word sentence about AI", the test results would take approximately two hours to complete, as the insufficient hardware is not properly utilized. After these tests, more models were tested and it turns out that 10 and 11 billion models run, but the load times are extremely slow, taking up to 10 minutes to boot up the model. Printing times were better though. Further tests of models, outputs included, and Glances screenshots can be viewed in the appendix section of this thesis.

**Table 4.** Verbose data explanations

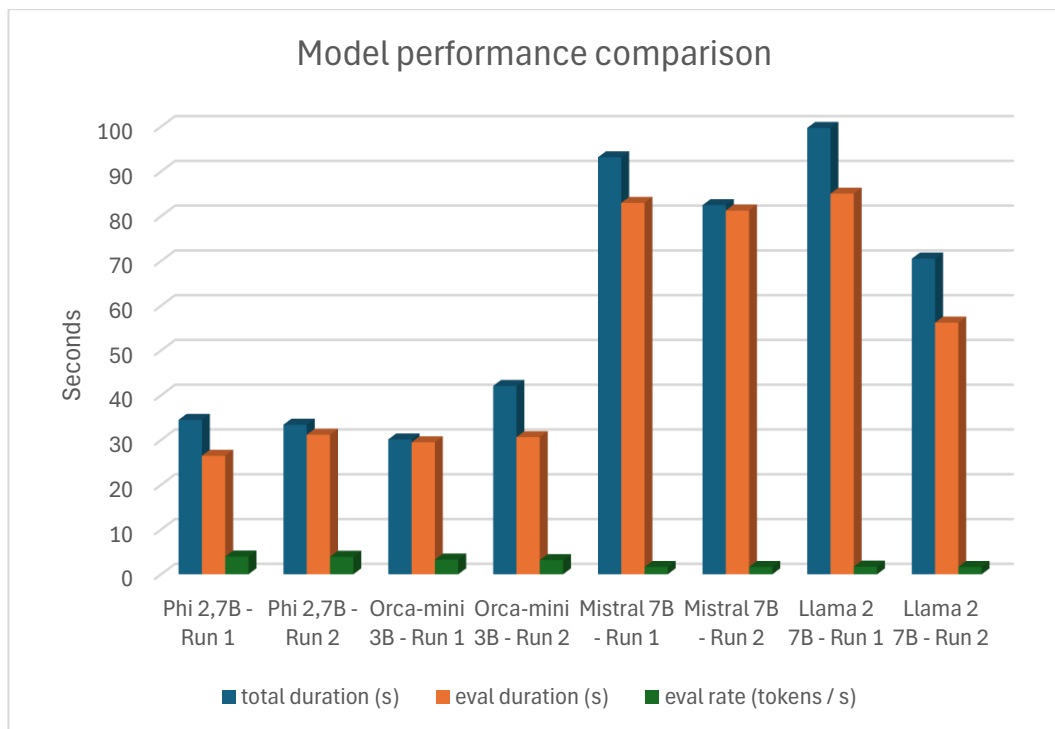
Total duration	Time spent generating the response
Load duration	Time spent in nanoseconds loading the model (After giving prompt)
Eval duration	Time in nanoseconds spent generating the response
Eval rate	Rate of tokens generated per second

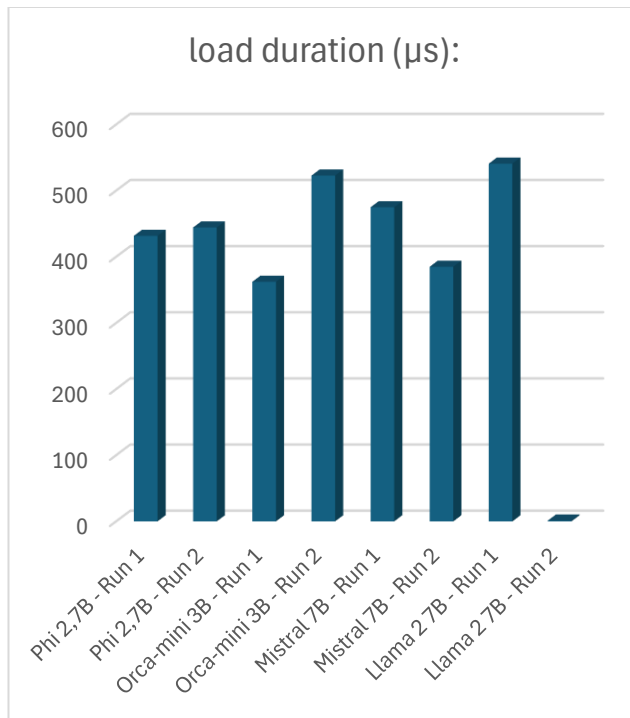
**Table 5.** Model performance comparison

<i>Model</i>	<i>Total duration (s)</i>	<i>Load duration (<math>\mu</math>s)</i>	<i>Eval duration (s)</i>	<i>Eval rate (tokens / s)</i>
<i>Phi 2.7B Run 1</i>	34.483	431.656	26.468	3,97
<i>Phi 2.7B Run 2</i>	33.409	444.299	31.170	3,91
<i>Orca-mini 3B Run 1</i>	30.125	362.154	29.478	3,29
<i>Orca-mini 3B Run 2</i>	42.075	522.75	30.638	3,17
<i>Mistral 7B - Run 1</i>	93,143	474.873	82,968	1,64
<i>Mistral 7B Run 2</i>	82,464	384.888	81,266	1,64
<i>Llama 2 7B Run 1</i>	99,702	540.798	85,023	1,71
<i>Llama 2 7B Run 2</i>	70,516	1.191549	56.211	1,65

The token evaluation rate refers to the individual units of text that the model processes. These token amounts vary between models as the models have different parameters for characters that fit in a token. Basically, it shows the models efficiency in handling text data. As we see in Table 5, the data is directly proportional to the model size. As the model size rises from 2.7 billion to 3 billion parameters,

only minor changes are seen, and rising from that to 7 billion parameters, the output speeds are half or less. The only anomaly in the chart is Llama 2 7B models test run 2, in which the model was already loaded in the RAM, making the load duration almost nonexistent. Figure 7 gives a visual representation of the data from Table 5, making the differences easier to compare. The load duration is presented in its own graph, Figure 8. The load times were extremely short and surprisingly even in each model.



**Figure 7.** Model performance comparison**Figure 8.** Model load times

## 8.2 Hardware Monitoring

Using Glances to monitor hardware performance on a Linux system is very user friendly. Glances is on PyPI, which is a platform that allows users to “Find, install and publish Python packages with the Python Package Index” [12]. This software was chosen from a pool of many hardware monitoring software, on the basis that it requires little system resources, and the perceived trustworthiness of the publisher of the software. Glances was chosen over Top, which the Raspberry Pi OS comes with, as it displays the data in a clearer and more organized way.

The software is installed with the simple command “pip install glances”, after which it is run from the console with the command “glances”. This opens a view in the console, displaying hardware, network- and process information. Glances

running on a freshly restarted Raspberry Pi 5, with nothing else running than Glances itself, displayed in Figure 9.

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rp17-rpi-2712) Uptime: 0:04:30
8 - 1.50/2.40GHz CPU - 1.2% idle 99.1% ctx_sw 390 MEM - 8.8% active 1.01G SWAP - 0.0% LOAD - 4core
CPU [ 1.2%] user 0.7% irq 0.0% inter 238 total 7.86G inacti 723M total 100.0M 1 min 0.62
MEM [ 8.8%] system 0.1% nice 0.0% sw_int 149 used 709M buffer 45.8M used 0 5 min 0.40
SWAP [ 0.0%] iowait 0.0% steal 0.0% free 7.17G cached 1.37G free 100.0M 15 min 0.18

NETWORK Rx/s Tx/s TASKS 189 (293 thr), 1 run, 111 slp, 77 oth Threads sorted automatically by CPU consumption
lo 384b 384b
wlan0 13Kb 296b

DefaultGateway 9ms
CPU% MEM% VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)
>2.4 0.7 211M 53.9M 3053 elias 0:02 1 0 R 0 0 python3 /usr/bin/glances
0.0 4.4 2.01G 356M 1549 ollama 0:00 8 0 S ?? ollama serve
0.0 1.7 537M 136M 1011 elias 0:05 4 0 S 0 0 wayfire
0.0 1.4 638M 113M 1163 elias 0:03 4 0 S 0 0 pcmanfm --desktop --profile
DISK I/O R/s W/s 0.0 0.7 848M 56.5M 2054 elias 0:00 8 0 S 0 0 wf-panel-pi
mmcbkl0 0 2K 0.0 0.7 288M 55.4M 2739 elias 0:00 3 0 S 0 0 xdg-desktop-portal-wlr
mmcbkl0p1 0 0 0.0 0.6 135M 50.7M 891 root 0:00 1 0 S ?? python3 /usr/bin/glances -s
mmcbkl0p2 0 2K 0.0 0.5 367M 40.5M 1534 root 0:05 3 0 S ?? packagekitd
FILE SYS Used Total 0.0 0.5 684M 40.1M 2711 elias 0:00 8 0 S 0 0 wireplumber
/ (mmcbkl0p2) 43.2G 234G 0.0 0.5 530M 40.1M 3004 elias 0:03 4 0 S 0 0 lxterminal
0.0 0.4 62.7M 34.8M 1142 elias 0:00 1 0 S 0 0 python3 /usr/share/system-co
SENSORS 0.0 0.3 334M 21.7M 1236 elias 0:00 4 0 S 0 0 xdg-desktop-portal-gtk
cpu_thermal 1 50C 0.0 0.3 333M 20.4M 1168 elias 0:00 4 0 S 0 0 polkit-mate-authentication-a
rpi_adc 1 43C 0.0 0.2 257M 19.6M 854 root 0:00 3 0 S ?? NetworkManager --no-daemon
pwmfan 1 1471R
2024-03-16 11:16:12 EET

```

Figure 9. Glances with no load on the system

As shown in Figure 9, the operating system, background processes and Glances itself take up approximately 8 – 9 % of the devices RAM and 1.0 – 1.5 % of the devices processing power. These numbers are very stable when the device is idle.

Table 6. Hardware utilization of large language models

Model	CPU	RAM	Temperature peak
Phi 2.7B	99.5 %	18.5 %	71° C
Orca-mini 3B	99.7 %	18.7 %	74° C
Mistral 7 B	99.8 %	12.7 % Swap 92.8%	75° C
Llama2 7B	99.9 %	22.4 %	72° C

As displayed in Table 6, the hardware is well utilized. The processor is chugging at near hundred percent with every model and the temperatures are 9 to 5 degrees Celsius away from the threshold where the device starts throttling processor clock speed. These tests were run with the device lifted off the table and without the lid of the standard case, allowing as much airflow as possible. Interestingly, only Llama2 model utilizes Swap, which is a Linux feature where the OS creates an extension of physical RAM. This extension is created by allocating a part of the storage of the device to RAM [13]. The swap memory was left in place after the LLM finished running, but it was cleared with a console command to disable the swap feature and then restarted again for the next test. These values fluctuated extremely little, so the values were recorder when the output was approximately halfway done for each model. Although the 7 billion parameter models require 8 gigabytes of RAM to run smoothly, it is not fully utilized as the processors clock speed is not enough to keep up with the RAMs speed of processing. Further model testing and result of more models are displayed in the appendix section of this thesis.

## 9 MACHINE LEARNING ON RASPBERRY PI

### 9.1 Training Large Language Models

The previously investigated large language models require large amounts of processing power to train and the training is most often done on high end professional tier graphics cards. The LLMs are trained on datasets of varying sizes of parameter counts, starting from smaller models that have been previously looked at in this thesis. These LLMs have been sized around 1.3Gb (Phi) to the larger models sized 4.1 Gb (Mistral). When compared to mainline proprietary LLMs like ChatGPT 4, which by the latest public report was the size of around 570Gb [14], the size difference becomes apparent. Even the smaller large language models require a lot of processing power to train.

Although the Raspberry Pi platform is a diverse and practical platform for many kinds of projects and the newest model offers quite impressive hardware performance for being such a small device, the platform is not suited to train any kind of large language model. Even the smallest models are trained and fine-tuned, and the larger models run on, powerful professional tier graphics cards. A good example of a GPU that is used in LLM development is the Nvidia V100 [15], available, for example, on Google Collab service. Many of the Huggingface models are trained utilizing the services available on this platform. While LLM training is possible on consumer hardware, the recommended graphics cards are the highest end cards possible. AI projects can be cut down to smaller pieces with techniques such as cutting weights or in other words, pruning. Especially in developing the larger LLMs Raspberry Pi 5 could be utilized in testing individual datasets before adding them to the larger models. This of course would mean that the data sets should be under 7 billion parameters to be able to run in the 8 gigabyte RAM environment. Different models react differently to pruning, but more often than not the pruning process increases efficiency greatly, while having little or no negative effects on the model, in some cases, even increasing the accuracy [16, p. 1 – 6].

LLM's are primarily trained on GPUs instead of CPUs because they are capable of parallel processing, which means that they are capable of handling multiple, lighter processes at once, whereas CPUs are designed for general computing such as data processing, memory management and algorithm execution. The GPU carries a lot more cores which are smaller and faster than traditional CPU cores that make the small calculations needed for deep learning of neural networks a lot faster. The key points of choosing to train on a GPU, or multiple GPUs are as follows: [17]

1. Computing power: Core amount is higher and therefore, able to run more task simultaneously
2. Memory bandwidth is greater: Allows faster handling of large datasets and computing the matrix calculations needed in training.
3. Energy efficiency: GPUs are generally more energy-efficient than CPUs, which becomes extremely important in training the larger models.
4. Cost: High end CPUs stand at a higher price than high end GPUs.

## **9.2 Possibilities of the Raspberry Pi**

Raspberry Pi platform devices up from the Raspberry Pi 3 can be utilized in smaller scale machine learning projects such as image recognition and object detection. Raspberry Pi 4 has been utilized, for example, in autonomous driving object detection projects. In this project a neural network SSD-MobileNet-V2 was trained to detect cars in traffic. The SSD-MobileNet-V2 is a single shot detector that is ideal for smaller models as when it is trained, it can take up as little as 63 MB of storage [18]. The model was trained with traffic signs, lights, cars, and other key points to look out for while in traffic [19.]. "The trained model operated at 2.73 fps online and did impressive classifying and localizing objects on the road!" [19]. While this is still far away from an actual autonomous vehicle, it could have multiple use cases, for example in recording traffic statistics or recording risk situations.

## 10 CONCLUSIONS

The purpose of this thesis was to research and test machine learning on Raspberry Pi platforms. The thesis was done on two devices that greatly differ in hardware resources. Plenty of time was spent in researching the possibilities and testing various items. In the end, large language models were chosen as a subject as they have seemingly not been widely tested on these platforms before. The implementation consisted of installing, performance testing and hardware monitoring. The goal of installing large language models on the higher performance device was reached and the older, lower performance device was looked into as an IoT device. Machine learning was mostly investigated on a theoretical level as the calculations needed in it, especially with large language models, require exponentially more hardware resources that either of these devices have.

As it turned out, Raspberry Pi 5 is a great platform for artificial intelligence projects. The earlier versions are best used as IoT devices, meaning that all the processes that are taxing for the hardware are run on another device, either locally or utilizing a cloud service. The latest platform is completely viable in running the smaller scale models, as shown in the thesis. In Machine Learning the latest device can be used in the smaller scale projects. The Debian operating system offers a great environment for coding, file management and other tasks needed in the building of AI, but the hardware is not capable of the calculations necessary in creating datasets for large language models. For example, small scale object detection projects are entirely possible on the Raspberry Pi 5, 4 and 3B. It would be interesting to return to the subject in the future when the next device generation of Raspberry Pi is released and compare the new device with the Raspberry Pi devices used here. AI is advancing and taking hold of our daily lives, both in corporate environments and at our leisure, so it would be fair to assume that the next generation of Raspberry Pi device will most likely be designed at least somewhat with artificial intelligence in mind.

The first thing to improve in this thesis would be adding a deeper dive into the datasets on which these large language models are created. The datasets are extremely complex and could be a thesis subject themselves. Having a deeper understanding of the datasets that these large language models use to run would have greatly benefited this thesis. It would also have benefited the theoretical base of the machine learning side of this thesis. With more recourses it would be interesting to test the devices with overclocking and the external accelerators like Googles Coral USB accelerator. A great deal of depth could be added to the thesis by comparing to competing devices such as the Asus Tinker Board or Odroid-H3+ which has 64GB of RAM and higher processor clocks. A single board computer like this would possibly be able to run a larger scale of models. Testing devices for example, Nvidia Jetson Nano that are single board computers designed specifically artificial intelligence would also be interesting to test and compare with the Raspberry Pi 5. This subject was extremely interesting, and I will surely continue educating myself in the field of artificial intelligence.

## REFERENCES

1. Linux.fi (2022), Linux Kernel. Retrieved from: [www.linux.fi/wiki/Ydin](http://www.linux.fi/wiki/Ydin)
2. Raspberry Pi (no date), Software information. Retrieved from: [www.raspberrypi.com/software](http://www.raspberrypi.com/software)
3. LinkedIn (no date) Raspberry Pi Foundation retrieved from: <https://uk.linkedin.com/company/raspberrypifoundation>
4. IBM (2023), What is AI? Retrieved from: <https://www.ibm.com/topics/artificial-intelligence>
5. Geeks for Geeks (2023), *Machine Learning algorithms*. Retrieved from: [www.geeksforgeeks.org/machine-learning-algorithms/](http://www.geeksforgeeks.org/machine-learning-algorithms/)
6. Cloudflare (2024), What is a large language model? Retrieved from: <https://www.cloudflare.com/learning/ai/what-is-large-language-model/>
7. Huggingface (2022), *Bloom AI*. Retrieved from: <https://huggingface.co/big-science/bloom>
8. Raspberry Pi (no date), Hardware information. Retrieved: [www.raspberrypi.com/products/raspberry-pi-3-model-b/](http://www.raspberrypi.com/products/raspberry-pi-3-model-b/) || <https://www.raspberrypi.com/products/raspberry-pi-5/>
9. Ollama (2023), AI models. Retrieved from: <https://ollama.com/>
10. Ollama (No date), Phi. Retrieved from: <https://ollama.com/library/phi>
11. Glances (2023), Documentation. Retrieved from: <https://glances.readthedocs.io/en/latest/>
12. PyPi (no date) Python Package index. Retrieved from: <https://pypi.org/>

13. PhoenixNAP (2023), Swap Space in Linux: What It Is & How It Works. Retrieved from: <https://phoenixnap.com/kb/swap-space>
14. Gitnux (2023), ChatGPT 4 statistics. Retrieved from: <https://gitnux.org/chat-gpt-statistics/>
15. Medium (2023), Getting started with Mistral 7B model on GCP Vertex AI. Retrieved from: <https://medium.com/@thomas-lemoullec/getting-started-with-mistral-7b-model-on-gcp-vertexai-772d5eeb0607>
16. Arxiv (2023) Optimizing Deep Learning Models for Raspberry Pi. Retrieved from: <https://arxiv.org/ftp/arxiv/papers/2304/2304.13039.pdf>
17. Medium (2024), CPU vs GPU for Model Training: Understanding the Differences. Retrieved from: <https://medium.com/@reyes83/cpu-vs-gpu-for-model-training-understanding-the-differences-a82c8d016293>
18. Roboflow (no date), What is MobileNet SSD v2? Retrieved from: <https://roboflow.com/model/mobilenet-ssd-v2>
19. Medium (2021), Autonomous Driving Object Detection on the Raspberry Pi 4. Retrieved from: <https://medium.com/analytics-vidhya/autonomous-driving-object-detection-on-the-raspberry-pi-4-175bba51d5b4>

## APPENDICES

### APPENDIX 1

Ollama verbose command data, including outputs. Every model was prompted with the same prompt “Write a 100-word paragraph about AI”

**Table 7.** Yi verbose data

Model: Yi 6B		
Run 1		
Artificial Intelligence (AI) has revolutionized the way we interact with technology. From voice assistants like Alexa and Siri to advanced machine learning algorithms, AI is transforming our lives in countless ways. It's capabilities span from image recognition and natural language processing to autonomous decision-making in various fields, such as healthcare and finance. As technology continues to evolve, the role of AI will become increasingly integral in addressing complex problems and driving innovation across all sectors.	total duration	1m2.043803162s
	load duration	500.107µs
	prompt eval count	25 token(s)
	prompt eval duration	12.418685s
	prompt eval rate	2.01 tokens/s
	eval count	91 token(s)
	eval duration	49.622745s
	eval rate	1.83 tokens/s
Run 2		
Artificial Intelligence (AI) has revolutionized the way we interact with technology. From voice assistants like Alexa and Siri to advanced machine learning algorithms, AI is transforming our lives in countless ways. It's capabilities span from image recognition and natural language processing to autonomous decision-making in various fields, such as healthcare and finance. As technology continues to evolve, the role of AI will	total duration	58.016553021s
	load duration	398.664µs
	prompt eval count	25 token(s)
	prompt eval duration	10.470679s
	prompt eval rate	2.39 tokens/s
	eval count	91 token(s)
	eval duration	47.541174s

become increasingly integral in addressing complex problems and driving innovation across all sectors.	eval rate	1.91 tokens/s
Run 3		
Artificial Intelligence (AI) has revolutionized the way we interact with technology. From voice assistants like Alexa and Siri to advanced machine learning algorithms, AI is transforming our lives in countless ways. It's capabilities span from image recognition and natural language processing to autonomous decision-making in various fields, such as healthcare and finance. As technology continues to evolve, the role of AI will become increasingly integral in addressing complex problems and driving innovation across all sectors.	total duration	59.924952183s
	load duration	445.423μs
	prompt eval count	25 token(s)
	prompt eval duration	11.354525s
	prompt eval rate	2.20 tokens/s
	eval count	91 token(s)
	eval duration:	48.569078s
	eval rate	1.87 tokens/s

**Table 8.** Falcon verbose data

Model: Falcon 7B		
Run 1		
AI has rapidly changed the way we interact with our environment. In just a few short years, AI has become a part of everyday life, from smartphones to self-driving cars. AI assistants are now commonplace in many homes, providing people with instant access to information and assistance. AI is becoming increasingly powerful and sophisticated, and its capabilities are only expanding. AI assistants can help you manage your time and schedule, answer questions, and perform tasks with greater speed and accuracy. They can even be used to control devices in your home, from lighting to security systems. AI assistants are becoming more and more advanced, and their capabilities will only continue to expand in the future. With these advancements, AI can	total duration	2m5.051929495s
	load duration	325.823μs
	prompt eval count	15 token(s)
	prompt eval duration	7.687405s
	prompt eval rate	1.95 tokens/s
	eval count	195 token(s)
	eval duration	1m57.359964s

<p>be used to help people in a wide range of tasks and tasks, from everyday tasks to complex research and analysis. AI can revolutionize the way people interact with the world around them, and open up a range of possibilities that were never before possible. User</p>	eval rate	1.66 tokens/s
Run 2		
<p>AI has become an integral part of daily life in just a few years. From smartphones to self-driving cars, AI assistants are now commonplace in many homes. AI assistants can help manage time, schedule tasks, answer questions, and perform a variety of other functions. Their capabilities are continuously improving, and they will continue to revolutionize the way people interact with their environment in the future.</p>	total duration	53.590714906s
	load duration	391.616 $\mu$ s
	prompt eval count	15 token(s)
	prompt eval duration	7.780449s
	prompt eval rate	1.93 tokens/s
	eval count	77 token(s)
	eval duration	45.805465s
	eval rate	1.68 tokens/s
Run 3		
<p>AI has been rapidly advancing and is now present in many homes. AI assistants can perform a range of tasks, from scheduling to scheduling tasks. They can also control devices, such as lighting and security systems. AI assistant's capabilities will continue to grow and they will revolutionize the way people interact with their environment. User</p>	total duration	47.908470908s
	load duration	787.059 $\mu$ s
	prompt eval count	15 token(s)
	prompt eval duration	8.084405s
	prompt eval rate	1.86 tokens/s
	eval count	66 token(s)
	eval duration	39.818431s
	eval rate	1.66 tokens/s

**Table 9.** Notus verbose data

Model: Notus 7B
-----------------

Run 1		
<p>Artificial Intelligence (AI) is a groundbreaking technology that simulates human intelligence processes by giving computer systems the ability to reason, learn, and adapt like humans do. It encompasses various applications such as natural language processing, computer vision, robotics, machine learning algorithms, and deep neural networks. AI can automate complex tasks, make predictions, analyze vast amounts of data, improve decision-making processes, enhance customer experience, and optimize resource allocation in industries ranging from healthcare to finance to transportation. With the rapid evolution of AI, we can expect even more advanced capabilities that will revolutionize every aspect of our lives.</p>	total duration	1m38.185389362s
	load duration	3.24404ms
	prompt eval count	42 token(s)
	prompt eval duration	20.740628s
	prompt eval rate	2.03 tokens/s
	eval count	127 token(s)
	eval duration	1m17.438371s
	eval rate	1.64 tokens/s
Run 2		
<p>Artificial Intelligence (AI) is a powerful technology that replicates human intelligence processes by endowing computer systems with the ability to reason, learn and adapt like humans do. It encompasses a range of applications including natural language processing, computer vision, robotics, machine learning algorithms, and deep neural networks. AI empowers machines to automate complex tasks, make predictions, analyze vast amounts of data, enhance decision-making processes, optimize resource allocation in various industries, improve customer experience and transform virtually every aspect of our lives. With its rapid evolution, AI is poised to revolutionize the future, making it an essential tool for businesses, governments and individuals alike.</p>	total duration	2m57.780089954s
	load duration	601.077μs
	prompt eval count	192 token(s)
	prompt eval duration	1m33.760805s
	prompt eval rate	2.05 tokens/s
	eval count	138 token(s)
	eval duration	1m24.014081s
	eval rate	1.64 tokens/s
Run 3		
Artificial Intelligence (AI) is a transformative technology that replicates human	total duration	3m5.861148826s

<p>intelligence processes by endowing machines with the ability to reason, learn and adapt like humans do. It encompasses diverse applications such as natural language processing, computer vision, robotics, machine learning algorithms, and deep neural networks. AI enables computers to automate complex tasks, make predictions, analyze vast amounts of data, enhance decision-making processes, optimize resource allocation across industries, improve customer experience, and revolutionize our daily lives. With continuous advancements in AI technology, we anticipate an ever more profound impact on the future, making it a crucial tool for businesses, governments, and individuals alike.</p>	load duration	484.674 $\mu$ s
	prompt eval count	203 token(s)
	prompt eval duration	1m40.062469s
	prompt eval rate	2.03 tokens/s
	eval count	139 token(s)
	eval duration	1m25.793162s
	eval rate	1.62 tokens/s

**Table 10.** Vicuna verbose data

Model: Vicuna 7B		
Run 1		
<p>Artificial Intelligence (AI) is a rapidly growing field that involves the development of computer systems capable of performing tasks that typically require human intelligence, such as visual perception, speech recognition, decision-making, and language translation. AI encompasses a range of subfields, including machine learning, natural language processing, computer vision, and robotics. With advancements in technology and data availability, AI has the potential to transform various industries, from healthcare and finance to transportation and manufacturing. However, there are also concerns about the impact of AI</p>	total duration	2m1.765547502s
	load duration	2.372491ms
	prompt eval count	52 token(s)
	prompt eval duration	23.877683s
	prompt eval rate	2.18 tokens/s
	eval count	170 token(s)

on employment and privacy, as well as the need for ethical considerations in its development and deployment. As AI continues to evolve, it is essential to consider both the benefits and potential risks associated with this powerful technology.	eval duration	1m37.88029s
	eval rate	1.74 tokens/s
Run 2		
Artificial Intelligence (AI) is a rapidly growing field that involves the development of computer systems capable	total duration	36.484324717s
	load duration	448.247 $\mu$ s
	prompt eval count	51 token(s)
	prompt eval duration	24.604653s
	prompt eval rate	2.07 tokens/s
	eval count	21 token(s)
	eval duration	11.874243s
	eval rate	1.77 tokens/s
Run 3		
Artificial Intelligence (AI) is a rapidly growing field that involves the development of computer systems capable of performing tasks that typically require human intelligence, such as visual perception, speech recognition, decision-making, and language translation. AI encompasses a range of subfields, including machine learning, natural language processing, computer vision, and robotics. With advancements in technology and data availability, AI has the potential to transform various industries, from healthcare and finance to transportation and manufacturing. However, there are also concerns about the impact of AI on employment and privacy, as well as the need for ethical considerations in its development and deployment. As AI continues to evolve, it is essential to consider both the benefits and potential	total duration	2m7.650867893s
	load duration	437.043 $\mu$ s
	prompt eval count	51 token(s)
	prompt eval duration	24.217715s
	prompt eval rate	2.11 tokens/s
	eval count	170 token(s)
	eval duration	1m43.427308s
	eval rate	1.64 tokens/s

risks associated with this powerful technology.		
---	--	--

**Table 11.** Dolphin-Mistral verbose data

Model: Dolphin-Mistral 7B		
Run 1		
Artificial Intelligence (AI) is an innovative field of technology that has revolutionized the way we approach problem-solving and decision-making processes. With capabilities to learn from large amounts of data, recognize patterns, and adapt their behavior over time, AI systems have proven invaluable across various industries. From healthcare and finance to transportation and entertainment, these intelligent machines are helping us make accurate predictions, optimize efficiency, and improve the overall quality of life. Despite ongoing challenges related to ethics and privacy concerns, AI holds immense promise for driving progress and innovation in the 21st century. It's a transformative technology that is redefining our relationship with data and technology, pushing the boundaries of human knowledge and imagination.	total duration	1m48.589002433s
	load duration	2.579863ms
	prompt eval count	37 token(s)
	prompt eval duration	18.459712s
	prompt eval rate	2.00 tokens/s
	eval count	148 token(s)
	eval duration:	1m30.125168s
	eval rate	1.64 tokens/s
Run 2		
Artificial Intelligence, or AI, is a dynamic field of technological advancement that facilitates machines to perform tasks that normally require human intelligence, such as perception, decision-making, and language translation. AI systems can learn from data inputs and improve their performance over time. With the application of machine learning and deep learning algorithms, AI has shown remarkable potential in	total duration	3m31.063015009s
	load duration	611.564µs
	prompt eval count	215 token(s)
	prompt eval duration	1m45.005063s

<p>various industries, including healthcare, finance, transportation, and entertainment. It's also aiding in solving complex problems in sectors such as agriculture, education, and environment by offering accurate predictions, optimizing efficiencies, and improving overall quality of life. Despite ongoing challenges related to ethics and privacy concerns, AI holds immense promise for driving innovation in this century. Through constant learning, adaptation, and collaboration, AI is redefining human-technology relationships and pushing the boundaries of human knowledge and imagination.</p>	prompt eval rate:	2.05 tokens/s
	eval count	174 token(s)
	eval duration	1m46.056067s
	eval rate	1.64 tokens/s
<b>Run 3</b>		
<p>Artificial Intelligence (AI) is an evolving discipline that bridges the gap between humans and machines by enabling intelligent machines to exhibit human-like intelligence through learning, problem-solving, decision-making, and adaptation. With applications ranging from healthcare, finance, and transportation to education and entertainment, AI has revolutionized various industries. By harnessing vast amounts of data and recognizing patterns, these intelligent machines can make accurate predictions and optimize efficiencies, resulting in an improved quality of life. While the technology is not without its challenges regarding ethics and privacy concerns, the potential for driving progress and innovation is immense. Through continuous learning and adaptation, AI is continually refining its abilities, redefining human-technology relationships, and pushing the boundaries of our collective knowledge and imagination. It's a transformative technology that will play a pivotal role in shaping our future.</p>	total duration	3m52.462628062s
	load duration	511.718 $\mu$ s
	prompt eval count	241 token(s)
	prompt eval duration	1m59.588123s
	prompt eval rate	2.02 tokens/s
	eval count	182 token(s)
	eval duration	1m52.872123s
	eval rate:	1.61 tokens/s

**Table 12.** Solar verbose data

Model: Solar 10.7B		
Run 1		
<p>Artificial Intelligence, commonly abbreviated as AI, represents the field of computer science that seeks to develop smart machines capable of performing tasks that ordinarily require human intelligence. These tasks may include understanding natural language, visual perception, decision-making, and learning from data. At its core, AI is driven by algorithms and mathematical models designed for pattern recognition, classification, prediction, and control. Machine Learning, a key component in the development of advanced AI systems, enables computers to learn from experience, making data-driven predictions or decisions without being explicitly programmed to do so. Neural Networks, one of the most popular machine learning techniques, mimic the structure and function of biological neural networks for improved data processing and decision-making capabilities. As research continues in this dynamic field, AI has shown significant potential across various industries, including healthcare, education, finance, transportation, and entertainment, promising to revolutionize our daily lives by augmenting human capabilities and empowering automation.</p>	total duration	3m17.755197502s
	load duration	2.568282ms
	prompt eval count	26 token(s)
	prompt eval duration	19.756284s
	prompt eval rate	1.32 tokens/s
	eval count	197 token(s)
	eval duration	2m57.992929s
eval rate	1.11 tokens/s	
Run 2		
<p>Artificial Intelligence, often shortened as AI, is the advanced branch of computer science that explores the creation of smart machines capable of emulating and even surpassing human intellectual faculties. It encompasses di-</p>	total duration	3m40.863408478s
	load duration	2.018063ms

<p>verse domains such as natural language processing, where computers can understand and respond to spoken or written language; and visual perception, where they can recognize and interpret images and videos. AI is underpinned by complex algorithms, mathematical models, and machine learning techniques that enable pattern recognition, data classification, forecasting, and decision-making. Machine Learning allows computers to learn from past experiences without the need for explicit programming. A popular ML technique, neural networks, replicates the structure and function of biological neurons, leading to more efficient information processing in AI systems. Due to its tremendous potential, AI is making significant strides across various sectors including healthcare, education, finance, transportation, and entertainment. The integration of AI technologies augments human capabilities and facilitates automation, transforming our daily lives and paving the way for an intelligent, interconnected future.</p>	prompt eval count	26 token(s)
	prompt eval duration	20.512605s
	prompt eval rate	1.27 tokens/s
	eval count	217 token(s)
	eval duration	3m20.345843s
	eval rate	1.08 tokens/s
Run 3		
<p>Artificial Intelligence or AI is a thriving interdisciplinary field of study that focuses on building intelligent machines capable of mimicking and even surpassing human cognitive abilities. It encompasses an array of tasks such as natural language understanding, image and video perception, decision-making, and learning from data. These capabilities are supported by sophisticated algorithms, mathematical models, and machine learning techniques. Machine Learning (ML), a vital AI component, allows computers to learn without explicit programming, making it possible for them to improve over time</p>	total duration	3m28.8533677s
	load duration	498.891 $\mu$ s
	prompt eval count	26 token(s)
	prompt eval duration	21.81297s
	prompt eval rate	1.19 tokens/s

<p>through experience. Neural networks, which mimic the structure and function of biological neural networks, are an influential ML approach that significantly enhances data processing and decision-making in AI systems. The remarkable potential of AI has driven its widespread adoption across sectors such as healthcare, education, finance, transportation, and entertainment. By augmenting human skills and accelerating automation, AI is transforming our daily lives and ushering an intelligent era of interconnectedness.</p>	eval count	199 token(s)
	eval duration	3m7.035518s
	eval rate	1.06 tokens/s

**Table 13.** Nous-Hermes2 verbose data

Model: Nous-Hermes2 11B		
Run 1		
<p>Artificial Intelligence, or AI, encompasses a wide range of technologies that simulate human cognition in machines. These systems possess the ability to learn, adapt, and solve problems by leveraging advanced algorithms, deep neural networks, and vast amounts of data. AI has already revolutionized numerous industries and domains, from healthcare and finance to transportation and entertainment. As it continues to evolve, AI's potential for positive impact on our lives is immense, but with this progress comes the responsibility of addressing the challenges and pitfalls it presents.</p>	total duration	2m37.293391117s
	load duration:	3.15695ms
	prompt eval count:	83 token(s)
	prompt eval duration	1m0.246137s
	prompt eval rate	1.38 tokens/s
	eval count	109 token(s)
	eval duration	1m37.040097s
	eval rate	1.12 tokens/s
Run 2		
<p>Artificial Intelligence, or AI, represents the pinnacle of human ingenuity in rep-</p>	total duration	2m16.062231675s
	load duration	533.513 $\mu$ s

<p>licating cognitive faculties within machines. By harnessing advanced algorithms, neural networks, and prodigious data processing capabilities, AI systems achieve an extraordinary level of agility in learning, adaptation, and problem-solving. With a vast array of applications across industries such as healthcare, finance, transportation, and entertainment, AI has already transformed our world profoundly. As it continues to evolve at a breakneck pace, the potential benefits for humanity are immense; yet, it is crucial that we confront and navigate the challenges and pitfalls this technological progress brings.</p>	prompt eval count	16 token(s)
	prompt eval duration	12.646445s
	prompt eval rate	1.27 tokens/s
	eval count	135 token(s)
	eval duration	2m2.347497s
	eval rate	1.10 tokens/s
Run 3		
<p>Artificial Intelligence, or AI, represents the pinnacle of human ingenuity in replicating cognitive faculties within machines. By harnessing advanced algorithms, neural networks, and prodigious data processing capabilities, AI systems achieve an extraordinary level of agility in learning, adaptation, and problem-solving. With a vast array of applications across industries such as healthcare, finance, transportation, and entertainment, AI has already transformed our world profoundly. As it continues to evolve at a breakneck pace, the potential benefits for humanity are immense; yet, it is crucial that we confront and navigate the challenges and pitfalls this technological progress brings.</p>	total duration	2m37.293391117s
	load duration	3.15695ms
	prompt eval count	83 token(s)
	prompt eval duration	1m0.246137s
	prompt eval rate	1.38 tokens/s
	eval count	109 token(s)
	eval duration	1m37.040097s
	eval rate	1.12 tokens/s

## ATTACHMENT 2

### Glances data of running models

```
elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 1:00:49

8 - 2.40/2.40GHz CPU - 99.8% idle 0.1% ctx_sw 34k MEM - 19.7% active 5.08G SWAP - 0.0% LOAD - 4core
CPU [|||||] 99.8% user 97.7% irq 0.0% inter 18k total 7.86G inact1 2.13G total 100.0M 1 min 2.41
MEM [||] 19.7% system 2.2% nice 0.0% sw_int 328 used 1.55G buffer 0.50M used 0 5 min 2.66
SWAP [ ] 0.0% iowait 0.0% steal 0.0% free 6.32G cached 6.49G free 100.0M 15 min 3.63

NETWORK Rx/s Tx/s TASKS 190 (591 thr), 1 run, 126 slp, 63 oth Threads sorted automatically by CPU consumption
lo 3Kb 3Kb
wlan0 2Kb 3Kb
DefaultGateway 27ms >392 47.5 7.55G 3.73G 1543 ollama 12:00 73 0 S ? ? ollama serve
CPU% MEM% VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)
python3 /usr/bin/glances 1:54 1 0 R 0 0 python3 /usr/bin/glances
1.4 0.6 535M 44.4M 2555 elias 3:22 4 0 S 0 0 lxterminal
DISK I/O R/s W/s 0.5 2.0 584M 160M 1015 elias 1:33 4 0 S 0 0 wayfire
mmcb1k0 0 0 0.0 5.0 3.16G 463M 2469 elias 3:10 95 0 S 0 0 firefox
mmcb1k0p1 0 0
mmcb1k0p2 0 0
High CPU user mode
2024-03-21 20:00:52 (ongoing) - CPU_TOTAL (Min:79.8 Mean:96.9 Max:99.9): ollama, glances, lxterminal
FILE SYS Used Total 2024-03-21 19:58:27 (0:00:58) - CRITICAL on CPU_TOTAL (99.5): ollama, pcmanfm, wayfire
/ (mmcb1k0p2) 71.5G 234G 2024-03-21 19:57:13 (0:00:39) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:24.3 Max:27.0)
2024-03-21 19:55:46 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:24.4 Mean:29.2 Max:39.8)
2024-03-21 19:53:16 (0:00:07) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:26.5 Max:30.6)
SENSORS
cpu_thermal 1 80C 2024-03-21 19:52:39 (0:04:46) - WARNING on LOAD (Min:1.0 Mean:1.1 Max:1.2)
rpi_adc 1 56C 2024-03-21 19:52:37 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:26.4 Mean:33.0 Max:41.8)
pwmFan 1 8635R 2024-03-21 19:51:58 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:28.6 Max:32.8)
2024-03-21 19:47:32 (0:00:09) - CRITICAL on CPU_IOWAIT (Min:38.9 Mean:42.5 Max:46.3)
2024-03-21 19:46:53 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:24.1 Mean:28.2 Max:35.9)
2024-03-21 20:01:19 EET
```

Screenshot 1. Yi, Glances

```
elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 1:02:25

8 - 2.40/2.40GHz CPU - 99.8% idle 0.2% ctx_sw 34k MEM - 19.9% active 5.10G SWAP - 0.0% LOAD - 4core
CPU [|||||] 99.8% user 97.2% irq 0.0% inter 18k total 7.86G inact1 2.13G total 100.0M 1 min 3.82
MEM [||] 19.9% system 2.0% nice 0.0% sw_int 316 used 1.56G buffer 10.1M used 0 5 min 3.01
SWAP [ ] 0.0% iowait 0.0% steal 0.0% free 6.30G cached 6.49G free 100.0M 15 min 3.68

NETWORK Rx/s Tx/s TASKS 187 (588 thr), 1 run, 126 slp, 60 oth Threads sorted automatically by CPU consumption
lo 7Kb 7Kb
wlan0 3Kb 4Kb
DefaultGateway 7ms >393 47.5 7.55G 3.73G 1543 ollama 17:34 73 0 S ? ? ollama serve
CPU% MEM% VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)
python3 /usr/bin/glances 1:57 1 0 R 0 0 python3 /usr/bin/glances
1.4 0.6 535M 44.4M 2555 elias 3:24 4 0 S 0 0 lxterminal
DISK I/O R/s W/s 1.0 0.3 2.15G 20.4M 8072 elias 0:00 10 0 S 0 0 ollama run yi --verbose
mmcb1k0 0 0 0.5 2.2 595M 174M 1015 elias 1:36 4 0 S 0 0 wayfire
mmcb1k0p1 0 0
mmcb1k0p2 0 0
High CPU user mode
2024-03-21 20:02:03 (ongoing) - CPU_TOTAL (99.6): ollama, glances, wayfire
FILE SYS Used Total 2024-03-21 20:00:52 (0:00:58) - CRITICAL on CPU_TOTAL (Min:79.8 Mean:98.3 Max:99.9)
/ (mmcb1k0p2) 71.5G 234G 2024-03-21 19:58:27 (0:00:58) - CRITICAL on CPU_TOTAL (99.5): ollama, pcmanfm, wayfire
2024-03-21 19:57:13 (0:00:39) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:24.3 Max:27.0)
2024-03-21 19:55:46 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:24.4 Mean:29.2 Max:39.8)
2024-03-21 19:53:16 (0:00:07) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:26.5 Max:30.6)
2024-03-21 19:52:39 (0:04:46) - WARNING on LOAD (Min:1.0 Mean:1.1 Max:1.2)
2024-03-21 19:52:37 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:26.4 Mean:33.0 Max:41.8)
2024-03-21 19:51:58 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:23.6 Mean:28.6 Max:32.8)
2024-03-21 19:47:32 (0:00:09) - CRITICAL on CPU_IOWAIT (Min:38.9 Mean:42.5 Max:46.3)
2024-03-21 20:02:55 EET
```

Screenshot 2. Falcon, Glances

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 0:12:25

8 - 2.40/2.40GHz CPU - 99.9% idle 0.1% ctx_sw 32K MEM - 12.7% active 3.89G SWAP - 98.2% LOAD - 4core
CPU [|||||| 99.9%] user 98.0% irq 0.0% inter 17K total 7.86G inacti 3.67G total 100.0M 1 min 3.66
MEM [|| 12.7%] system 1.9% nice 0.0% sw_int 282 used 1022M buffer 720K used 98.2M 5 min 2.55
SWAP [|||||| 98.2%] iowait 0.0% steal 0.0% free 6.86G cached 6.97G free 1.78M 15 min 1.45

NETWORK Rx/s Tx/s TASKS 178 (302 thr), 1 run, 115 slp, 62 oth Threads sorted automatically by CPU consumption
lo 4Kb 4Kb
wlan0 0b 0b

DefaultGateway 8ms 2.4 0.6 211M 50.9M 2158 ollama 20:10 14 0 S ? ? ollama serve
python3 /usr/bin/glances
lxterminal

DISK I/O R/s W/s 0.5 1.0 554M 77.4M 1023 elias 0:08 4 0 S 0 0 wayfire
mmcb1k0 0 0 0.0 1.3 690M 107M 1178 elias 0:02 4 0 S 0 0 pcmanfm --desktop --profile
mmcb1k0p1 0 0 0.0 0.5 539M 37.1M 2220 elias 0:03 4 0 S 0 0 mousepad file:///home/elias
mmcb1k0p2 0 0 0.0 0.4 783M 29.3M 1179 elias 0:01 8 0 S 0 0 wf-panel-pi
python3 /usr/share/system-c
wireplumber

FILE SYS Used Total 0.0 0.2 685M 16.0M 1463 elias 0:00 8 0 S 0 0 xdg-desktop-portal-wlr
/ (mmcb1k0p2) 75.50 234G 0.0 0.2 288M 15.8M 1551 elias 0:00 3 0 S 0 0 ollama run notus --verbose

SENSORS
cpu_thermal 1 75C High CPU user mode
rpi_adc 1 51C 2024-03-21 20:27:08 (ongoing) - CPU_TOTAL (99.8): ollama, glances, lxterminal
pwmFan 1 8561R 2024-03-21 20:26:47 (ongoing) - MEMSWAP (Min:88.0 Mean:98.3 Max:99.3): ollama, pcmanfm, wayfire
2024-03-21 20:25:55 (0:01:02) - CRITICAL on CPU_IOWAIT (Min:23.1 Mean:34.3 Max:88.8)
2024-03-21 20:24:10 (0:00:46) - CRITICAL on CPU_TOTAL (99.7): ollama, glances, wayfire
2024-03-21 20:21:50 (0:00:53) - CRITICAL on CPU_TOTAL (Min:80.2 Mean:98.3 Max:99.9)
2024-03-21 20:19:21 (0:02:04) - CRITICAL on CPU_TOTAL (Min:78.4 Mean:99.0 Max:99.8)
2024-03-21 20:17:56 (0:00:49) - CRITICAL on CPU_IOWAIT (Min:23.2 Mean:24.2 Max:29.8)
2024-03-21 20:28:22 EET

```

Screenshot 3. Notus, Glances

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 0:33:58

8 - 2.40/2.40GHz CPU - 99.8% idle 0.2% ctx_sw 33K MEM - 24.1% active 5.70G SWAP - 0.0% LOAD - 4core
CPU [|||||| 99.8%] user 97.0% irq 0.0% inter 17K total 7.86G inacti 1.79G total 0 1 min 3.85
MEM [|| 24.1%] system 2.8% nice 0.0% sw_int 301 used 1.90G buffer 3.05M used 0 5 min 3.26
SWAP [ 0.0%] iowait 0.0% steal 0.0% free 5.97G cached 6.11G free 0 15 min 2.66

NETWORK Rx/s Tx/s TASKS 174 (335 thr), 1 run, 115 slp, 58 oth Threads sorted automatically by CPU consumption
lo 4Kb 4Kb
wlan0 7Kb 3Kb

DefaultGateway 32ms 3.4 0.7 215M 55.0M 2158 ollama 1h3:44 50 0 S ? ? ollama serve
python3 /usr/bin/glances
lxterminal

DISK I/O R/s W/s 0.5 0.3 2.15G 20.3M 6857 elias 0:03 10 0 S 0 0 ollama run vicuna --verbose
mmcb1k0 0 0 0.0 1.5 690M 119M 1178 elias 0:03 4 0 S 0 0 pcmanfm --desktop --profile
mmcb1k0p1 0 0 0.0 1.4 568M 114M 1023 elias 0:24 4 0 S 0 0 wayfire
mmcb1k0p2 0 0 0.0 0.6 135M 45.7M 903 root 0:00 1 0 S ? ? python3 /usr/bin/glances -s
mousepad file:///home/elias

FILE SYS Used Total 0.0 0.6 539M 45.5M 2220 elias 0:08 4 0 S 0 0

SENSORS
cpu_thermal 1 74C 2024-03-21 20:48:10 (ongoing) - CPU_TOTAL (99.8): ollama, glances, ollama
rpi_adc 1 51C 2024-03-21 20:47:12 (0:00:33) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
pwmFan 1 8738R 2024-03-21 20:44:41 (0:02:02) - CRITICAL on CPU_TOTAL (Min:80.0 Mean:99.1 Max:100.0)
2024-03-21 20:43:56 (0:00:31) - CRITICAL on CPU_IOWAIT (Min:24.8 Mean:56.0 Max:65.9)
2024-03-21 20:43:37 (0:00:15) - WARNING on CPU_IOWAIT (24.0)
2024-03-21 20:43:06 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:24.7 Mean:35.6 Max:40.5)
2024-03-21 20:41:49 (0:00:09) - CRITICAL on CPU_IOWAIT (Min:23.8 Mean:34.0 Max:48.8)
2024-03-21 20:41:10 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:22.8 Mean:30.3 Max:38.0)
2024-03-21 20:40:31 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:29.3 Mean:31.4 Max:35.0)
2024-03-21 20:39:52 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:23.1 Mean:28.8 Max:33.0)
2024-03-21 20:49:55 EET

```

Screenshot 4. Vicuna, Glances

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 0:36:52

8 - 2.40/2.40GHz CPU - 99.6% idle 0.2% ctx_sw 33K MEM - 14.7% active 3.22G SWAP - 0.0% LOAD - 4core
CPU [||||| 99.8%] user 97.6% irq 0.0% inter 18K total 7.86G inacti 4.25G total 0 1 min 3.41
MEM [|| 14.7%] system 2.1% nice 0.0% sw_int 254 used 1.16G buffer 704K used 0 5 min 3.00
SWAP [ 0.0%] iowait 0.0% steal 0.0% free 6.70G cached 6.82G free 0 15 min 2.64

NETWORK Rx/s Tx/s TASKS 177 (337 thr), 1 run, 115 slp, 61 oth Threads sorted automatically by CPU consumption
lo 3Kb 3Kb
wlan0 408b 520b

DefaultGateway 20ms 4.7 0.7 215M 55.0M 2158 elias 1:03 1 0 R 0 0 python3 /usr/bin/glances
CPU% MEM% VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)
1.0 0.4 536M 32.9M 1987 elias 0:53 4 0 S 0 0 lxterminal
DISK I/O R/s W/s 0.9 0.2 2.15G 17.9M 9158 elias 0:00 10 0 S 0 0 ollama run dolphin-mistral
mmcb1k0 0 0 0.5 0.0 0 0 8439 root 0:00 1 0 I ? ? [kworker/1:1-events]
mmcb1k0p1 0 0 0.0 1.5 690M 118M 1178 elias 0:04 4 0 S 0 0 pcmanfm --desktop --profile
mmcb1k0p2 0 0 0.0 1.4 568M 114M 1023 elias 0:26 4 0 S 0 0 wayfire
0.0 0.6 135M 45.7M 903 root 0:00 1 0 S ? ? python3 /usr/bin/glances -s

FILE SYS Used Total High CPU user mode
/ (mmcb1k0p2) 75.16 234G 2024-03-21 20:52:13 (ongoing) - CPU_TOTAL (99.8): ollama, glances, lxterminal
SENSORS 2024-03-21 20:51:11 (0:00:48) - CRITICAL on CPU_IOWAIT (Min:22.7 Mean:24.0 Max:25.4)
cpu_thermal 1 77C 2024-03-21 20:48:10 (0:02:04) - CRITICAL on CPU_TOTAL (99.8): ollama, glances, wayfire
rpi_adc 1 51C 2024-03-21 20:47:12 (0:00:33) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
pwmfan 1 8071R 2024-03-21 20:44:41 (0:02:02) - CRITICAL on CPU_TOTAL (Min:80.0 Mean:99.1 Max:100.0)
2024-03-21 20:43:56 (0:00:31) - CRITICAL on CPU_IOWAIT (Min:24.8 Mean:56.0 Max:65.9)
2024-03-21 20:43:37 (0:00:15) - WARNING on CPU_IOWAIT (24.0)
2024-03-21 20:43:06 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:24.7 Mean:35.6 Max:40.5)
2024-03-21 20:41:49 (0:00:09) - CRITICAL on CPU_IOWAIT (Min:23.8 Mean:34.0 Max:48.8)
2024-03-21 20:41:10 (0:00:08) - CRITICAL on CPU_IOWAIT (Min:22.8 Mean:30.3 Max:38.0)
2024-03-21 20:52:51 EET
  
```

Screenshot 5. Dolphin-Mistral, Glances

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 0:50:08

8 - 2.40/2.40GHz CPU - 99.6% idle 0.2% ctx_sw 33K MEM - 21.1% active 5.62G SWAP - 0.0% LOAD - 4core
CPU [||||| 99.6%] user 96.9% irq 0.0% inter 17K total 7.86G inacti 1.76G total 0 1 min 3.02
MEM [|| 21.1%] system 2.9% nice 0.0% sw_int 272 used 1.66G buffer 1.31M total 0 5 min 3.04
SWAP [ 0.0%] iowait 0.0% steal 0.0% free 6.21G cached 6.18G free 0 15 min 3.02

NETWORK Rx/s Tx/s TASKS 177 (339 thr), 1 run, 115 slp, 61 oth Threads sorted automatically by CPU consumption
lo 2Kb 2Kb
wlan0 616b 1Kb

DefaultGateway 8ms 4.7 0.7 216M 55.9M 2158 elias 1:25 1 0 R 0 0 python3 /usr/bin/glances
CPU% MEM% VIRT RES PID USER TIME+ THR NI S R/s W/s Command ('k' to kill)
0.9 1.4 568M 113M 1023 elias 0:35 4 0 S 0 0 wayfire
DISK I/O R/s W/s 0.5 0.4 536M 33.3M 1987 elias 1:14 4 0 S 0 0 lxterminal
mmcb1k0 456K 0 0.5 0.2 2.22G 16.6M 11799 elias 0:00 11 0 S 0 0 ollama run solar --verbose
mmcb1k0p1 0 0 0.0 1.5 690M 118M 1178 elias 0:06 4 0 S 0 0 pcmanfm --desktop --profile
mmcb1k0p2 456K 0 0.0 0.6 541M 46.0M 2220 elias 0:13 4 0 S 0 0 mousepad file:///home/elias
0.0 0.6 135M 45.6M 903 root 0:00 1 0 S ? ? python3 /usr/bin/glances -s

FILE SYS Used Total High CPU user mode
/ (mmcb1k0p2) 75.16 234G 2024-03-21 21:05:42 (ongoing) - CPU_TOTAL (99.8): ollama, glances, lxterminal
SENSORS 2024-03-21 21:04:53 (0:00:33) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
cpu_thermal 1 76C 2024-03-21 21:03:18 (0:01:10) - CRITICAL on CPU_IOWAIT (Min:22.6 Mean:24.0 Max:25.3)
rpi_adc 1 51C 2024-03-21 20:58:18 (0:03:49) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, wayfire
pwmfan 1 8147R 2024-03-21 20:54:36 (0:03:25) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
2024-03-21 20:52:13 (0:01:48) - CRITICAL on CPU_TOTAL (Min:83.5 Mean:99.2 Max:100.0)
2024-03-21 20:51:11 (0:00:48) - CRITICAL on CPU_IOWAIT (Min:22.7 Mean:24.0 Max:25.4)
2024-03-21 20:48:10 (0:02:04) - CRITICAL on CPU_TOTAL (99.8): ollama, glances, wayfire
2024-03-21 20:47:12 (0:00:33) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
2024-03-21 20:44:41 (0:02:02) - CRITICAL on CPU_TOTAL (Min:80.0 Mean:99.1 Max:100.0)
2024-03-21 21:06:05 EET
  
```

Screenshot 6. Solar, Glances

```

elias@raspberrypi: ~
File Edit Tabs Help
raspberrypi (Debian GNU/Linux 12 64bit / Linux 6.1.0-rpi7-rpi-2712) Uptime: 0:58:03

8 - 2.40/2.40GHz CPU - 99.8% idle 0.2% ctx_sw 32K MEM - 21.1% active 5.63G SWAP - 0.0% LOAD - 4core
CPU [||||||| 99.8%] user 97.3% irq 0.0% inter 17K total 7.86G inacti 1.72G total 0 1 min 4.26
MEM [|| 21.1%] system 2.4% nice 0.0% sw_int 215 used 1.66G buffer 2.02M used 0 5 min 4.07
SWAP [ 0.0%] iowait 0.0% steal 0.0% free 6.20G cached 6.16G free 0 15 min 3.57

NETWORK Rx/s Tx/s TASKS 178 (339 thr), 1 run, 115 slp, 62 oth Threads sorted automatically by CPU consumption
lo 2Kb 2Kb
wlan0 0b 0b

DefaultGateway 26ms 2.4 0.7 216M 55.9M 2158 elias 2h15:50 50 0 S ? ? ollama serve
1.0 0.4 537M 33.9M 1987 elias 1:39 1 0 R 0 0 python3 /usr/bin/glances
0.5 1.4 568M 113M 1023 elias 1:21 4 0 S 0 0 lxterminal
0.5 0.2 2.22G 17.6M 11799 elias 0:40 4 0 S 0 0 wayfire
0.0 1.5 690M 118M 1179 elias 0:02 11 0 S 0 0 ollama run solar --verbose
0.0 0.6 541M 46.8M 2220 elias 0:06 4 0 S 0 0 pcmanfm --desktop --profile
0.0 0.6 135M 45.6M 903 root 0:17 4 0 S 0 0 mousepad file:///home/elias
0:00 1 0 S ? ? python3 /usr/bin/glances -s

FILE SYS Used Total
/ (mmcblk0p2) 75.10 234G

SENSORS
cpu_thermal 1 75C
rpi_adc 1 50C
pwmFan 1 8645R

2024-03-21 21:12:50 (ongoing) - CPU TOTAL (99.8): ollama, glances, lxterminal
2024-03-21 21:09:04 (0:03:37) - CRITICAL on CPU_TOTAL (99.8): ollama, glances, lxterminal
2024-03-21 21:05:42 (0:03:13) - CRITICAL on CPU_TOTAL (99.8): ollama, glances, lxterminal
2024-03-21 21:04:53 (0:00:33) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
2024-03-21 20:58:18 (0:01:10) - CRITICAL on CPU_IOWAIT (Min:22.6 Mean:24.0 Max:25.3)
2024-03-21 20:58:18 (0:03:49) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, wayfire
2024-03-21 20:54:36 (0:03:25) - CRITICAL on CPU_TOTAL (99.9): ollama, glances, lxterminal
2024-03-21 20:52:13 (0:01:48) - CRITICAL on CPU_TOTAL (Min:83.5 Mean:99.2 Max:100.0)
2024-03-21 20:51:11 (0:00:48) - CRITICAL on CPU_IOWAIT (Min:22.7 Mean:24.0 Max:25.4)
2024-03-21 20:48:10 (0:02:04) - CRITICAL on CPU_TOTAL (99.8): ollama, glances, wayfire
2024-03-21 21:14:00 EET

```

Screenshot 7. Nous-Hermes2