Exploring large vision-language models with prompt engineering for peripheral blood cell image analysis and classification.

Universitat Oberta de Catalunya

UNIVERSITAT DE BARCELONA

Marina Sánchez Quijada

Master in Bioinformatics and Biostatistics

Area:

Machine Learning

Name of the Advisor:

Edwin Santiago Alférez Baquero

18th of June of 2024

Title:	*Exploring large vision-language models with prompt engineering for peripheral blood cell image description and classification.*
Name of the author:	*Marina Sánchez Quijada*
Name of the advisor:	*Edwin Santiago Alférez Baquero*
SRP:	*Edwin Santiago Alférez Baquero*
Date of delivery:	*06/2024*
Studies:	Master in Bioinformatics and Biostatistics
Area:	*Machine Learning*
Language:	*English*
Keywords:	*Machine learning* *Medical imaging analysis* *Large visual-language models*

Abstract

In recent years, large vision and language models (LVLMs) have gained a lot of attention due to their accessibility and impressive performance in various language and vision tasks. Consequently, their applications in the medical imaging field are being studied, showing already great potential in clinical settings. However, very few studies have been carried out to evaluate the potential of LVLMs for disease diagnosis, especially for microscopy images. In this work, we explore for the first time the capabilities of three of the most advanced LVLMs (GPT-4, Claude3, and LLaVa) in the analysis and classification of peripheral blood cells.

To perform this exploration, we build multiple prompts based on different prompting techniques, including few-shot learning and chain of thought (CoT), to study and improve the performance of these LVLMs for blood cell image analysis. We also explore the functionality of the assistant and the system roles in model behaviour and performance. Moreover, we perform a comprehensive comparison of their accuracy rates and create a web application for white blood cell classification.

Our experiments conclude that the best-performing method and LVLM combination is GPT-4o when using a two-shot learning strategy with the addition of the assistant role. When testing this approach on 100 images of leukocytes, we attained an accuracy rate of 78%. Although this performance is not reliable enough and LVLMs should not be used as diagnostic tools, we believe that due to the rapid advancement of large language-vision models, LVLMs could become a great asset in the analysis of pathology images, working as an assistant for quick blood cell description and classification.

The code and prompts used in this work are accessible on GitHub.

Contents

1. Introduction

1.1. Context and justification of the work:

Medical images (microscopy images, X-rays, or magnetic resonance images) are commonly used by clinicians to extract information about the condition of patients, make diagnoses, and plan appropriate treatments. However, this type of assessment and analysis requires a lot of expertise, is very time-consuming, and is subjected to intraobserver variability and cognitive bias. Poor-resolution images, for instance, can lead to misleading observations and delayed or inappropriate treatment. Moreover, the huge increase in medical imaging data in recent years has made the management and analysis of these images a challenge.

Over the past few years, machine learning and deep learning algorithms have become an important topic in the field of medical imaging. These new techniques can facilitate the detection of subtle abnormalities not even visible to the human eye, mitigate cognitive bias, and help to perform faster and more objective analysis, thus improving the fidelity of diagnosis. For instance, some pretrained convolution neural networks (CNNs) and transformer-based models have shown great performance in disease diagnosis and prognosis (Cui et al. 2023).

More recently, large vision and language models (LVLMs) have gained a lot of attention because of their accessibility and impressive performance in various language and vision tasks. Although most of these models have not been fine-tuned on medical data, they are extensively trained on vast amounts of text and vision data and have already shown great potential in clinical applications (Yan et al. 2023). However, few attempts have been made to assess the capabilities of multimodal large vision-language models (LVLMs) in the medical image domain. In this work, we aim to evaluate the performance and robustness of GPT-4, Claude3 and LLaVa in the analysis of pathology images.

For this reason, we will focus on the description, morphological analysis, and classification of peripheral blood cell images. The observation of peripheral blood smears has been used by pathologists over the years as a major tool in the analysis, diagnosis, and monitoring of blood-related diseases, such as blood disorders or cancers like leukaemia, lymphomas, or myelomas. However, as we mentioned before, this technique can be quite subjective, as it is influenced by intra-observer variability and can be subjected to perceptual and interpretative errors. We propose the use of LVLMs to mitigate these issues, minimize misdiagnoses, and improve the time of analysis.

1.2. Objectives:

The main objective of this work is to explore the capabilities of large language-vision models for the morphological analysis and classification of blood cells. Hence, it is necessary to perform prompt engineering on these multimodal models, build prompts for the extraction and classification of blood cell morphology features, and compare the results obtained with different prompting techniques and LVLMs. Nonetheless, we intend to create an application web, using the model and prompts with the best performance, to serve as a tool for leukocyte classification.

Therefore, the objectives of this thesis can be broken down into the following items:

1. Research the state of the art of LVLMs in the medical image domain.
2. Obtain morphological descriptions of the peripheral blood cell types.
3. Create prompts for the description and classification of peripheral blood cells.
4. Compare the efficiency of different prompting techniques to analyse and classify peripheral blood cells.
5. Compare the capabilities of different large vision-language models to analyse and classify peripheral blood cells.
6. Build an application web for the classification of leukocytes.

1.3. Sustainability, social-ethical and diversity impact:

This section, included inside the Global and ethical commitment competence (GECC), addresses acting in an honest, ethical, sustainable, responsible, and respectful manner for human rights in academic and professional settings. It also includes the design of solutions to improve these practices.

Accordingly, this competence is divided into three different dimensions:

- Sustainability
- Ethics and social responsibility
- Diversity, genre, and human rights

In terms of sustainability, large language and vision models have a considerable environmental and carbon impact. This technology requires high-performance computers and GPUs, which require the mining of rare metals and the release of large amounts of greenhouse gasses. Additionally, these infrastructures are usually distributed in big computing facilities and, although their footprint may vary depending on where they are located, they use vast amounts of energy and water, and might also contribute to soil pollution (Rillig et al. 2023). A recent study has estimated that in 2027, the global AI may be responsible for up to 6.6 billion cubic metres of water withdrawal, more than half of the total annual water withdrawal of the United Kingdom (Li at el. 2023).

Another important issue surrounding large language models involves the data used to pretrain them. On the one hand, the datasets used can contain biases and discrimination against marginalised groups of people, which can be engraved into the machine learning algorithm. It is also important to note that most of the data used to train medical models come from high-income regions, so they may not represent the reality of low- or middle-income countries (Li et al. 2023). Therefore, it is important to identify these sources of potential bias moving forward.

On the other hand, training data used in the medical field can contain protected or sensitive health information that could interfere with the patient's right to privacy. By illicitly using this information, patient data could be exploited by third parties without their consent. Accordingly, there should be more transparency concerning data collection, storage, and usage. Moreover, data protection laws should be put in place to regulate their sale to private companies and ensure the ethical use and protection of this medical information (Chiruvella & Guddati, 2021). In our case, we will be working with images collected from blood smears of patients at the Hospital Clínic of Barcelona, of which no personal data nor medical history is known, ensuring patient privacy and confidentiality (Acevedo et al. 2020).

1.4. Approaches and methods:

As a first step, we will perform a literature search on the history and state of the art of LVLMs, as well as their applications in medical image processing. Then, we will explore the GPT-4, Claude3 and LLaVa models to become acquainted with them and learn how to perform prompt engineering. As we expressed before, we will be analysing microscopic images of peripheral blood cells, so we will research the distinctive morphological features of these cells, establish a classification table, and sort out a database to use for this project.

When all this has been established, we will perform prompt engineering to evaluate the performance of these LVLMs in the analysis of peripheral blood cell images. We will accomplish this by using their dedicated chat interface, as well as their officially released APIs. We will initiate our dialogue with both image and text inputs, asking for both image analysis and feature description, as well as cell type classification.

We will build multiple prompts based on different prompting techniques and learning methods, such as few-shot learning and chain of thought (CoT), to extract this information and evaluate their effectiveness. We will also introduce the assistant and system roles to the prompt, along with the user role, to test their functionality in model behaviour and performance. Additionally, we will establish a criterion to gauge the accuracy of the answers and compare the results between them. To eliminate any biases and avoid any interference from previous conversations, we will start a new session each time.

Following this, we will do a systematic comparison of the results obtained with the different models and prompting techniques to describe the most successful combinations and evaluate their capabilities as a tool in medical imaging. Finally, we will develop a web application for white blood cell classification using the best performing model and prompting strategy. This application will be created as a tool for researchers to easily analyse and classify blood cell images and to showcase the potential of these LVLMs.

1.5. Planification:

1.5.1. Tasks:

The work presented in this document will be divided into four phases, concurring with the four deliverables described in the subject guide:

- **Work plan**: in this first phase, we will define the project and plan out the tasks. Moreover, we will perform a first literature search to learn about the subject at hand, the state of the art of large multimodal models in the medical image domain, and learn how to work with GPT-4, Claude3 and LLaVa. The following table shows these tasks in a more comprehensive manner, as well as their estimated durations:

Table 1. Planification of the first part of the project (Work plan).

Tasks	Duration (days)
Project description and work planning	8
Bibliographic search of LVLMs and prompt engineering	6
Familiarization with GPT-4, Claude3 and LLaVa	3
PEC1 writing and delivery	3

- **Work development 1**: in this second stage, we will research a dataset of microscopic blood cell images and explore their distinctive morphological features. We will carry out a first approximation to prompt engineering, create prompts based on different learning and prompting techniques (zero-shot, few-shot learning, and chain of though), and test them on the LVLMs. We will explore different metrics to evaluate model performance and start comparing the capabilities of each model and technique.

Table 2. Planification of the second part of the project (Work development 1).

Tasks	Duration (days)
Examination of peripheral blood cell images	5
Creation of prompts to describe and classify cell images (1)	20
Comparison and evaluation of models and prompting techniques (1)	4
PEC2 writing and delivery	5

- **Work development 2**: in this second part of the development, we will continue testing the models to try to improve cell classification accuracy. Moreover, we will introduce the assistant and system roles to the previous prompts to explore their function in model behaviour and performance. We will continue doing a systematic comparison between them, to describe the finest LVLMs and techniques to classify peripheral blood cells. Finally, we will create a web application with the best performing LVLM and prompting strategy.

Table 3. Planification of the third part of the project (Work development 2).

Tasks	Duration (days)
Creation of prompts to describe and classify cell images (2)	20
Comparison and evaluation of models and prompting techniques (2)	4
Building of a web application for leukocyte classification	5
PEC3 writing and delivery	5

- **Final report and presentation**: in this last phase of the project, we will write the final report and design and record the presentation. Furthermore, we will create a GitHub repository with the code utilised during the thesis.

Table 4. Planification of the fourth part of the project (Final report and presentation).

Tasks	Duration (days)
Writing of the final memory	12
Creation of the GitHub repository	2
Preparation and recording of the presentation	6
PEC4 delivery	1

1.5.2. Calendar:

Here, we created a Gantt Chart to schedule the previous tasks and visualize their duration in a calendar. Each image shows a different phase of the project, with the activities on the left side and time intervals on the right side.

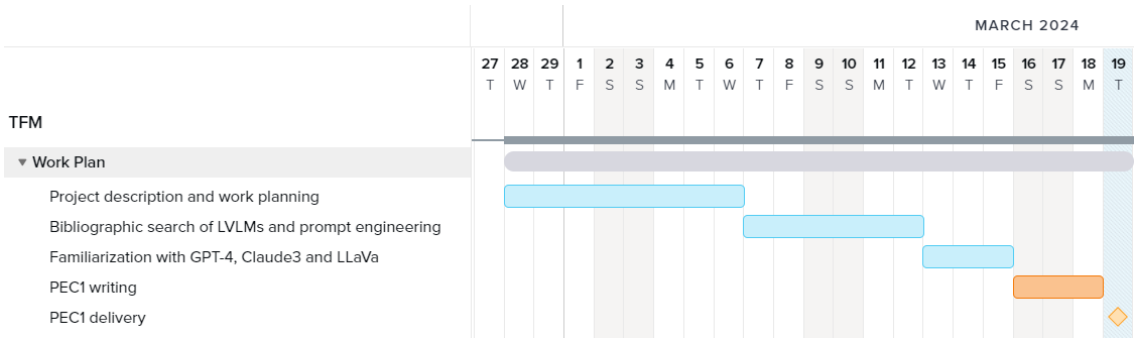


Figure 1. Gantt Chart of the first phase of the project (Work Plan)

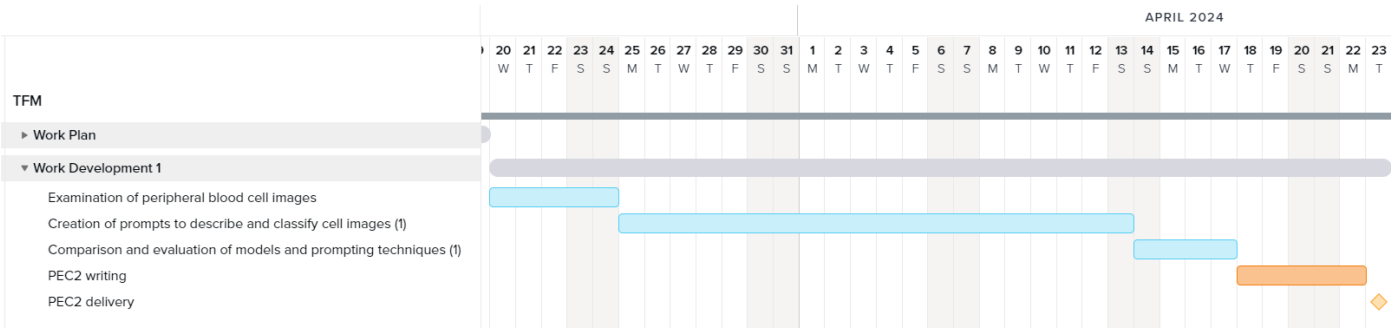


Figure 2. Gantt Chart of the second phase of the project (Work Development 1)

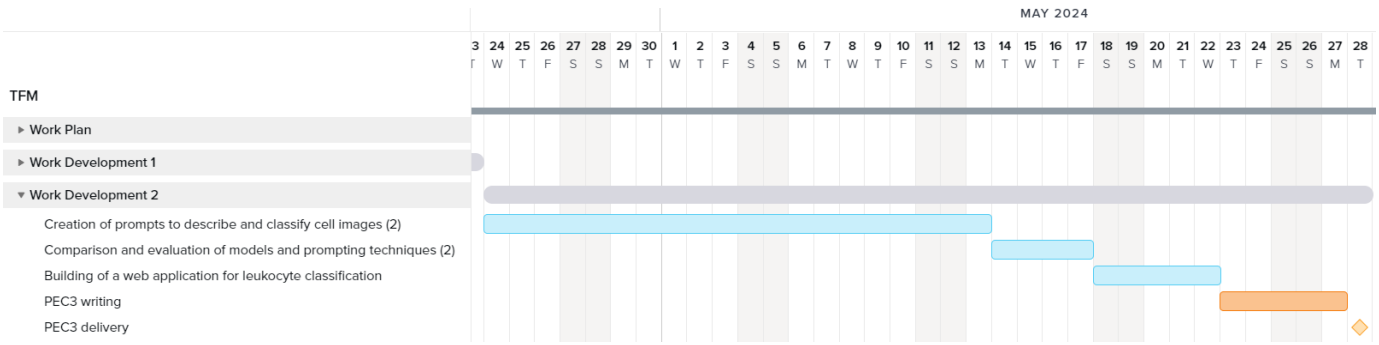


Figure 3. Gantt Chart of the third phase of the project (Work Development 2)

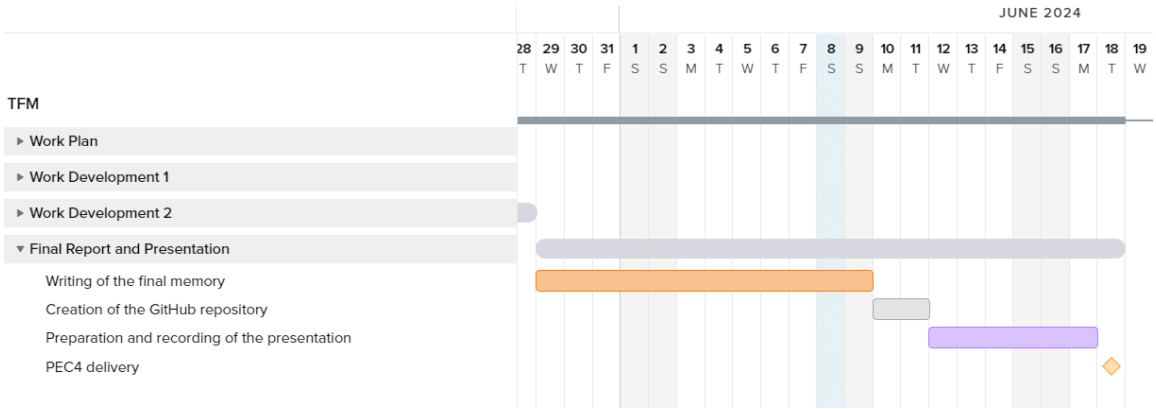


Figure 4. Gantt Chart of the fourth phase of the project (Final Report and Presentation)

1.5.3. Risk analysis:

The main risk of this project is the lack of time to develop it, as some of the previous tasks might take longer than expected. It is also important to keep in mind that time is a limiting factor during the week due to work and other obligations, and most of these tasks are going to be performed during the weekends.

Another limiting factor for the development of this project is the cost of LVLMs. The usage of GPT-4 and Claude3 have a cost associated, that responds to the type of model and the number of tokens used, both in the prompt and the output. Hence, we will have to adjust the number of images used for every cell type. Our ability to perform very complex and cost-demanding prompt techniques might also be limited. We will be testing Claude3's cheapest model, Haiku, and some more affordable alternatives to GPT-4V, such as GPT-4 Omni, to assess cost-effectiveness and to attempt to reduce expenses in the long run.

An additional constraint of these models are the rate and token limits, which can restrict both the number of API requests that we can make and the length of the prompts that we can create. For instance, we might not be able to generate prompts with a lot of sample images, which reduces our ability to perform few-shot learning.

Lastly, we must contemplate the availability and reliability of these LVLMs. We are depending on OpenAI and Anthropic for GPT-4 and Claude3 to be accessible and operational, and to provide reliable and timely responses. A failure of their infrastructure, service interruptions, or other errors in latency or maintenance might complicate our work.

1.6. Expected results:

By the end of the project, we want to accomplish the first-ever study of the capabilities of LVLMs in the analysis of stained microscopy blood cell images. Moreover, we expect to perform a comprehensive comparison of the effectiveness of different prompting techniques and LVLMs for this task and describe the most successful ones for their potential use in the medical imaging domain. As a final achievement, we want to create a web application capable of identifying and classifying leukocytes from peripheral blood cell smear images.

Accordingly, we expect to obtain the following items:

1. A final dissertation in line with the objectives and tasks we have presented in this planning.
2. A web application that uses LVLMs to classify white blood cell images.
3. A complete work plan of the project.
4. A virtual presentation of the final thesis.

2. Large language-vision models

2.1. Origin and functioning of LVLMs:

Large language and vision models have gained considerable popularity over the past couple of years, especially with the arrival of ChatGPT to the public sphere in 2022. However, to comprehend the origins and evolution of LVLMs, we must go back to the rise of natural language processing (NLP), a subfield of computer science that allows computers to interpret, comprehend and manipulate human language. Although this field dates to the 1940s, with World War II, it was not until the late 1980s with the increase in computer power and the advancements in machine learning that language models (LMs) started to flourish. LMs are an NLP technology that uses machine learning algorithms to analyse large amounts of text to understand and replicate human language styles and patterns. Another big breakout for NLP was the incorporation of deep learning methods, such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM), which provided LMs with a more nuanced comprehension of textual context, allowing the models to understand the context of words that were far away from one another (Carolan et al. 2024). However, the pivotal moment that led to the creation of large language models (LLMs) was the introduction of the Transformer architecture by Vaswani et al. in 2017. This architecture is based on attention mechanisms, which consist of assigning levels of importance to different words in a sentence to focus on the most important parts without relying on the distance between them. Since then, several successful LLMs have been created, like Google's BERT (Bidirectional Encoder Representations from Transformers) and AI's GPT (Generative Pre-trained Transformer).

Parallelly, vision models have evolved as a specialisation of language models, first designed to process and understand images. However, in later years, vision models have been developed to align both vision and text information in a multimodal manner, and have been used in a variety of applications, such as image captioning. Most large multimodal models are based on CLIP (Contrastive Language-Image Pre-Training), a multimodal approach trained on a wide range of text paired images found across the internet. By pretraining an image and a text encoder on these pairs, the model learns to recognize visual concepts in the images and associate them with their names, performing vision-related tasks without being specially instructed, thus in a zero-shot manner (Radford et al. 2021).

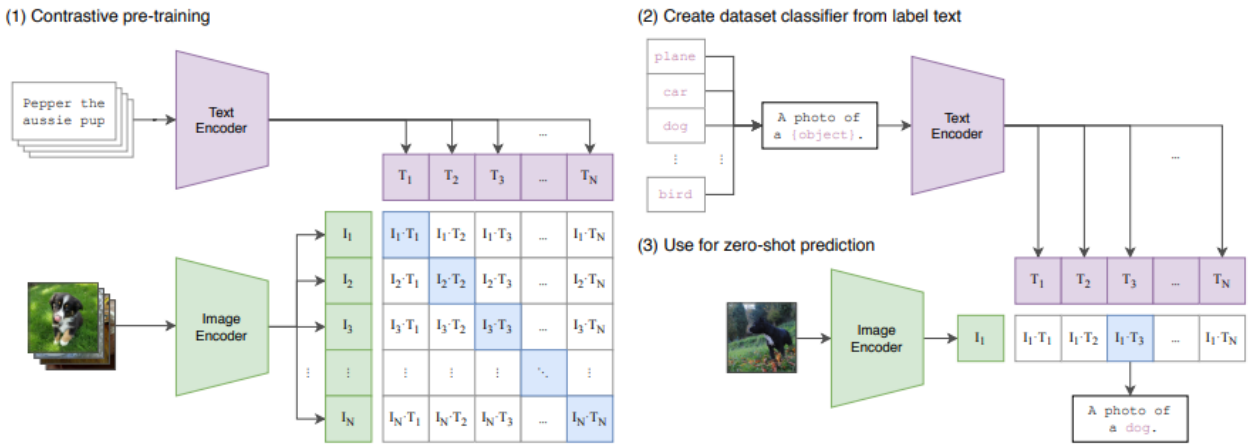


Figure 5. Overview of CLIP. The text and image encoders are pretrained on image-text pairs. When tested, the trained text encoder predicts the class of the caption that associates better with the given image, turning CLIP into a zero-shot classifier (Radford et al. 2021).

This simple approach is highly efficient and very flexible, and it does very well in recognizing common objects, as the model is trained on a wide range of visual concepts. However, it may struggle with images not covered in the pretraining dataset and with more abstract or systematic tasks.

2.2. Most advanced LVLMs:

Some of the most well-known, used and advanced large vision-language models are described below:

- **GPT-4**: this transformer-based model, created by OpenAI, is the fourth-generation language model in the GPT series. It is trained by both publicly available data and data licensed from third-party providers (Achiam et al. 2023). It is also the first of its series to contain multimodal capabilities, with the ability to accept and generate both text and image data. This new model shows an improved performance in NLP tasks in contrast to the previous GPT-3.5, scoring in the 90% percentile in the bar examen compared to human performance. It also has several advanced features compared to its predecessors, including an increased model size of 175 billion parameters, better model accuracy and performance, multilingual capabilities, better contextual understanding, and improved reasoning capabilities. However, GPT-4 also presents some limitations and challenges to be addressed. Due to the large scale of this model, it requires large computational resources to train and use effectively, making fine-tuning a challenge, especially for tasks with limited labelled data. Moreover, the interpretation of how the model reaches its decisions and output can be quite challenging (Baktash & Dawodi, 2023). Different

multimodal versions of this model can be found, including GPT-4V, GPT-4 Turbo, and the newest GPT4o, published on the 13th of May 2024, which is faster and more affordable than its predecessor.

- **Claude3**: published in 2023, Claude3 is a family of multimodal models created by Antrophic with sophisticated language and vision capabilities. It includes the models Opus, Sonnet, and Haiku, ranging from higher performance and complexity to more affordable and faster models. It is trained on a mixture of public, private and synthetic data, although its architecture is not available. According to Anthropic, their most intelligent model, Opus, outperforms GPT4 and Gemini in some of the most common evaluation benchmarks for AI systems, including undergraduate-level expert knowledge, graduate-level expert reasoning, and basic mathematics. In its vision capabilities, Claude stands on par with other state-of-the-art models, such as GPT-4V and Gemini Ultra (Anthropic, 2024).

- **Gemini**: created by Google in 2023, this family of large multimodal models includes capabilities for text, image, audio, and video. Three models can be found, with different sizes: Ultra, the largest and most capable model for complex tasks, Pro, the best model for scaling across a wide range of tasks, and Nano, the smallest and most efficient for on-device tasks. According to Google, at the moment of its release, Gemini's Ultra surpassed the state-of-the-art results for 30 of the 32 academic benchmarks used in LLM research. It also outperformed human experts on massive multitask language understanding (MMLU), which tests both world knowledge and problem-solving abilities on 57 subjects like math, medicine, physics or history. Moreover, it showed great performance in image, video, and audio processing, surpassing even GPT4-V in some benchmarks (Pichai & Hassabis, 2023).

- **LLaVa-1.6:** this large language and vision assistant, released as an open-sourced model, consists of a pretrained CLIP vision encoder and a large language model, trained end-to-end for both visual and language understanding purposes (Liu et al. 2023). This last LLaVa model has several improvements compared to LLaVA-1.5, maintaining its efficiency but with improved performance and more capabilities. The image input resolution has been increased to 4x more pixels, allowing the model to grasp more visual details. Moreover, it also has better visual reasoning, logical reasoning, and world knowledge. Regarding performance, it surpasses other open-sourced large multimodal models but not commercial ones, obtaining similar results to Gemini Pro on some

benchmarks. Where LLaVA-1.6 shines in its Chinese multimodal capabilities, surpassing every other model in benchmarks like MMBench-CN (Liu et al. 2024).

As mentioned before, the applications of these large vision and language models are large, including text-to-video generation, image captioning, text-to-speech, and more. It is no surprise that they have been utilised in a wide range of disciplines, such as business, education, social studies, politics, and even agriculture.

2.3. LVLMs in medical imaging:

2.3.1. Specialised models:

The potential of some LLMs, such as ChatGPT, has been previously explored in the biomedical and healthcare domain, performing biomedical text mining and even in the context of clinical practice and research. For instance, Sorin et al. 2023 explored the potential of ChatGPT-3.5 as a tool for facilitating clinical decision-making in breast tumour cases, with favourable results. Some LLMs have even been trained using large datasets of medical literature to create specialised models for the interpretation of patient information, generating medical reports or assisting healthcare professionals (Hartsock & Rasool, 2024). One of the best examples of this is Med-PaLM2, a large language model designed to provide high-quality answers to medical questions (Singhal et al. 2023), that reached an 86.5% accuracy on the MedQA medical exam benchmark.

However, the capabilities of LVLMs have been poorly explored for their use in medical imaging. Some attempts have been made with CT scans, MRIs, or X-rays, but not so much with microscopy images. Some specialized models trained for medical visual question answering (VQA), where the model is asked to provide an accurate response to a medical question posed about an image, are described below:

- LLaVa-Med: this multimodal model, capable of understanding, conversing, and assisting with inquiries about medical images, was created by fine-tuning the LLaVa model on a figure-caption dataset of 600,000 examples extracted from PubMed Central. Trained in less than 15 hours, this model exhibits excellent performance, outperforming other state-of-the-art models on certain metrics (Li et al. 2023).
- Visual Med-Alpaca: an open-source parameter-efficient biomedical foundation model created for multimodal biomedical tasks, such as interpreting radiological images. It was created by fine-tuning the LLaMa-7B model on a set of 54,000 medical question-answer pairs, curated by GPT-3.5-turbo and human experts (Shu et al. 2023)

- Med-Flamingo: this multimodal few-shot learner, based on the OpenFlamingo-9B architecture, was trained on a curated database of paired and interleaved medical image text data from publications and medical textbooks. This approach enables the model to generalize and perform diverse multimodal tasks with only a few examples (Moor et al. 2023)

Some other specialized models, focused on a specific type of medical image, include XrayGPT for chest radiograph summarization (Thawkar et al. 2023) and MedXChat for chest X-ray understanding (Yang et al., 2023), OphGLM with applications in ophthalmology (Gao et al. 2023), and RaDialog, a radiology report generation and conversational assistance model (Pellegrini et al. 2023).

2.3.2. Prompt engineering:

Apart from fine-tuning and the creation of specialized models, there is one other strategy used to try to improve model performance: prompt engineering. This in-context learning strategy involves enhancing an already pre-trained model with prompts, specifically task instructions, to steer the model's behaviour and output to the desired one. This is achieved by designing and optimizing these prompts. By using this simple and ingenue methodology, we can evaluate and harness the full potential of foundation models. Some of the most frequently used techniques to achieve more complex tasks and improve the performance of LVLMs are zero-shot prompting, few-shot prompting, and chain-of-thought.

Zero-shot prompting is used to evaluate the fundamental capabilities of these models when no examples or demonstrations are provided to the model. However, when this approach does not work, usually in more complex tasks, it is recommended to add some examples to the prompt. This approximation is known as few-shot learning and is used to soft-train the model and enable in-context learning. The examples provided guide the models to act in a certain way and to generate specific responses (Touvron et al. 2023). Depending on the number of examples we use, we will perform one-shot, two-shot, three-shot, five-shot, etc. However, these strategies may still provide unreliable or erroneous outputs, especially for more complex reasoning tasks. In this case, it is advised to experiment with more advanced prompting techniques, such as the popular chain of thought (CoT). CoT is used to improve the ability of LVLMs to perform complex reasoning through the introduction of a series of intermediate reasoning steps. This prompting strategy has been shown to improve model performance on a range of arithmetic, commonsense, and symbolic reasoning tasks (Wei et al. 2022). It is usually combined with few-shot prompting as demonstrated in Figure 6, although it can be used with zero-shot prompting by adding "Let's think step by step" to the original prompt (Kojima et al. 2022).

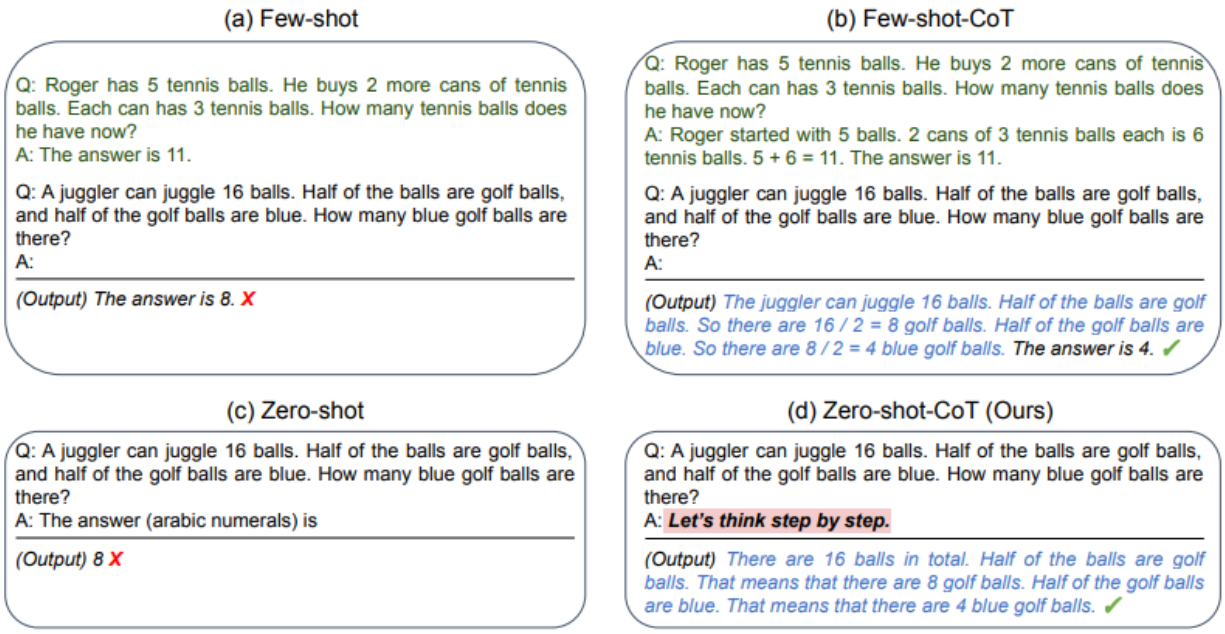


Figure 6. Example inputs and outputs with (a) Few-shot, (b) Few-shot CoT, (c) standard Zero-shot and (d) Zero-shot CoT. The blue text shows the multi-step reasoning generated by chain-of-thought prompting. (Kojima et al. 2022)

On the other hand, the messages or inputs provided to the LVLMs have two properties: the content, which contains the task or instructions given to the model, and the role. The latter can take one of three values: user, assistant, or system. The user role represents the input provided by the individual interacting with the AI, through giving prompts or asking questions to the model, while the assistant role delivers the model's response to the user's input. On the other hand, the system role is used to provide instructions to the model, give context for the conversation, or set behaviour rules. Integrating the assistant and system roles into the prompt can help set the context and direct the conversation, resulting in improved performance and enhanced inputs.

Some approaches have been made on this front to evaluate the potential of LVLMs for medical imaging, especially with ChatGPT and GPT models. For instance, a recent study on the capabilities of GPT-4V for VQA on radiology images using a zero-shot strategy found an accuracy rate of 50% for both open-ended and closed-ended questions (Yan et al. 2023). Likewise, Van et al. 2024 reported an accuracy of 61.82% and 51.61% when performing zero-shot learning with ChatGPT-4 on a CX-ray and a brain tumour RMI database, respectively. These researchers also obtained similar results when testing BiomedCLIP, OpenCLIP, OpenFlamingo, and LLaVa on the same databases. Nonetheless, the accuracy rate of OpenFlamingo increased significantly (71.28%) when few-shot learning was performed, adding four images from the CX-ray database to the prompt as examples to build some context.

However, most of the specialized models and explorations of LVLMs for medical imaging focus on X-ray, radiography, and magnetic resonance images, and very few studies have been made with microscopy pathological images. A recent article by

Yan et al. 2023 evaluated the zero-shot performance of GPT-4V, through the ChatGPT webpage version, on the PathVQA (Pathology Questions for Medical Visual Question Answering) database, which includes microscopy, dermoscopy, WSI and endoscopic videos. The overall accuracy score obtained in this instance was 29.9%. However, the accuracy of the QA pairs deemed “easy” by the researchers was set at 75%, showing proficiency in basic medical knowledge. However, for the hardest QA pairs, where the model was asked to give clinical advice, the score set was 8.03%. Another paper reported a surprisingly 84.85% accuracy rate for ChatGPT-4 when testing it on an acute lymphoblastic leukaemia image database, containing microscopic images of normal and blast cells (Van et al 2024).

3. Material and methods

3.1. Database:

For this work, we have selected a public dataset stored as a Mendeley repository named "A dataset for microscopic peripheral blood cell images for development of automatic recognition systems" (Acevedo et al. 2020). This dataset contains 17,092 images of individual normal cells, labelled by clinical pathologists as neutrophils, eosinophils, basophils, lymphocytes, monocytes, immature granulocytes, erythroblasts, and platelets. Neutrophils are differentiated into segmented and band neutrophils.

Images were captured in the Core Laboratory at the Hospital Clinic of Barcelona, with the analyser CellaVision DM96 after automated May Grünwald-Giemsa staining. Blood samples were obtained from healthy patients without infection or hematologic diseases. To protect the anonymity of these individuals, images were saved using random numbers.

The following figure shows images of the different types of cells contained in the database:

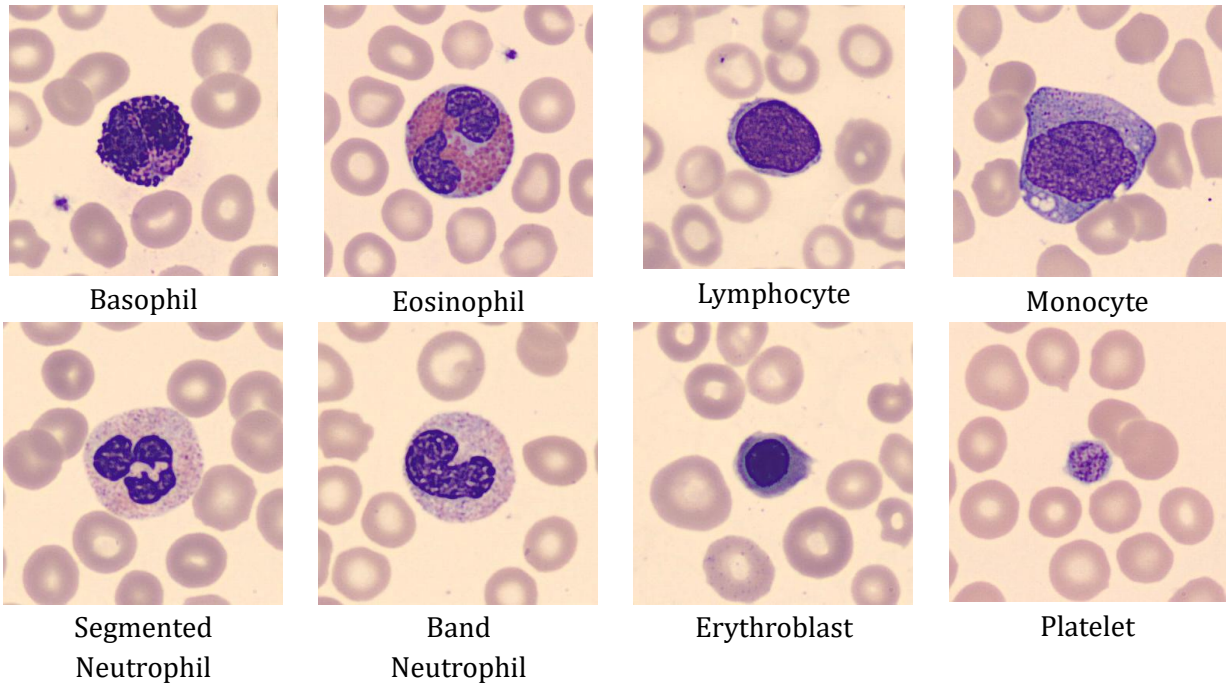


Figure 7. Images of the different blood cell types found in peripheral blood smears and provided by the database (Acevedo et al. 2020).

3.2. Prompt engineering strategies:

For the exploration of LVLMs, we adopted a zero-shot and few-shot approach, where we asked the models to answer questions about the peripheral blood cell images we provided. We tried with very straightforward and simple questions (basic prompts) or by asking the models for a step-by-step justification of their answers. We also performed different prompt engineering techniques, such as role prompting and chain of thought (CoT). Role prompting was used to assign a role to the LVLMs, give them some context, and make the AI act as if they were pathologists. On the other hand, CoT was used to promote more complex reasoning in the large vision-language models through intermediate reasoning steps. Lastly, we set the temperature of all models at 0 to make the answers as deterministic and reproducible as possible.

To perform zero-shot learning, we did not provide any previous context or examples to the model, so we could evaluate the fundamental capabilities of the models for blood cell image analysis and classification. We used a basic prompt, asking for the model to identify the blood cell type in the image, role prompting, asking for the same while making the model act as a "pathologist who specializes in the analysis of peripheral blood smears", and chain of thought. To perform CoT with zero-shot learning, we asked the model to provide a step-by-step reasoning before answering (Fig. 8).

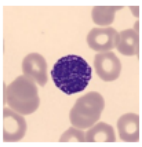
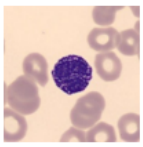
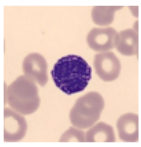
	Basic Prompt	Role Prompting	Zero-Shot CoT
USER:	Identify the blood cell type in this image: 	**You are a pathologist who specializes in the analysis of peripheral blood smears.** Identity the blood cell type in this image: 	Identity the blood cell type in this image. **Provide a step-by-step reasoning before providing your answer:** 
ASSISTANT OUPUT:	The cell in the center of this image appears to be a type of white blood cell known as a neutrophil...	The cell in the image appears to be a neutrophil, which is a type of white blood cell (leukocyte)...	To identify the type of blood cell in the image, we can follow these steps...

Figure 8. Prompt strategies used for zero-shot learning.

On the other hand, we carried out few-shot learning by soft-training the models in advance with sample images for each leukocyte type. For this exploration, we excluded platelets and erythroblasts due to the limitations in time and tokens. For this very reason, we tested the accuracy of the models after training them only with one, two, or three labelled examples for each cell type. Therefore, we performed zero-shot, two-shot, and three-shot learning, respectively. In addition to using a basic prompt, we also tested CoT, although this strategy changes slightly from zero-shot CoT. In this case, we provided the model with intermediate reasoning steps by

complementing the examples with reasoned answers describing the morphological characteristics of each cell type. These descriptions were obtained by performing a bibliographical search on the cell size, nucleus shape, nucleocytoplasmic ratio, cytoplasm colour, and granulation of the blood cell types. This information was collected in a table, which was fed as a CSV file to GPT-4. Then, we asked the model to generate descriptions for each cell type based on the features shown in the file. By using this technique, we were able to steer the model's behaviour to provide reasoned outputs. Moreover, we tried a third approach, by repeating CoT but explicitly asking the model for a step-by-step reasoning before answering.

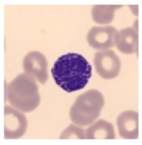
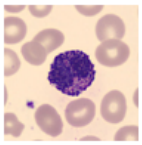
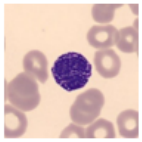
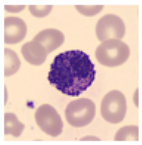
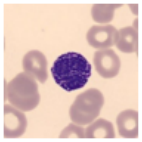
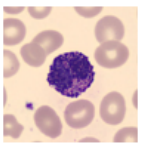
	Basic Prompt	CoT	CoT (Step-by-step)
USER:	Identify the blood cell type in this image:  This image features a basophil. (...) Identify the blood cell type in this image: 	Identify the blood cell type in this image:  This image features a basophil. **Basophils have an intermediate size, low nucleocytoplasmic ratio, and a segmented nucleus whose segments can vary. They contain dark, condensed/heterogeneous nuclear chromatin and a wide azurophilic cytoplasm with round basophilic granulation. The large number of dark purplish granules often make the nucleus difficult to see.** (...) Identify the blood cell type in this image: 	Identify the blood cell type in this image:  This image features a basophil. Basophils have an intermediate size, low nucleocytoplasmic ratio, and a segmented nucleus whose segments can vary. They contain dark, condensed/heterogeneous nuclear chromatin and a wide azurophilic cytoplasm with round basophilic granulation. The large number of dark purplish granules often make the nucleus difficult to see. (...) Identify the blood cell type in this image. **Provide a step-by-step reasoning before providing your answer:** 
ASSISTANT OUPUT:	This image features a basophil.	This image shows a neutrophil. Neutrophils are characterized by their intermediate size, low nucleocytoplasmic ratio, and a segmented nucleus with 2-5 lobes connected by thin chromatin strands...	To identify the blood cell type in this image, let's go through a step-by-step reasoning process: 1.Size and nucleocytoplasmic ratio: The cell appears to be small in size with a high nucleocytoplasmic ratio...

Figure 9. Prompt strategies used for few-shot learning. Only one example of a basophil is shown in this figure, although more sample images were included in the prompt where the ellipsis is located.

Finally, we used the accuracy as a metric to evaluate the model performance. The prompts were tested at first with 5 images per cell type, which were selected at random. For zero-shot learning, we included leukocytes, platelets, and erythroblasts, but to test few-shot learning only leukocytes were used, due to limitations in the number of tokens. To compare the performance between models and prompting strategies, the same 5 images were used in each instance. Moreover, to obtain more

robust results and better understand the behaviour of the best performing models, we tested them on 100 new peripheral blood cell images, featuring 20 basophils, 20 eosinophils, 20 lymphocytes, 20 monocytes and, 20 neutrophils (10 band neutrophils, and 10 segmented neutrophils).

On the other hand, only correct and specific answers were used to calculate the accuracy rate. For instance, answers that identified the cells as "leukocytes", were not clear enough or provided the user with more than one possible answer ("lymphocyte or monocyte") were not considered.

3.3. User, assistant and system roles:

In this work, we also explored the function of the user, assistant and system roles in model behaviour and performance. As a first approach, we tried the prompting strategy described in the previous section as a single user prompt. After this first attempt at evaluating model performance, we introduced the assistant role with few-shot learning. Instead of providing all examples in a single user message, we intercalated the user and assistant roles, as shown in Fig. 10. This way, we could recreate more accurately the structure and behaviour of a conversation between the individual and the AI.

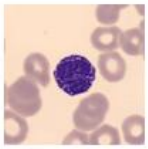
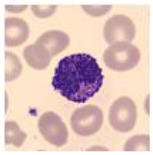
Basic Prompt	
USER:	Identify the blood cell type in this image: 
ASSISTANT:	This image features a basophil.
USER:	Identify the blood cell type in this image: 
ASSISTANT OUPUT:	This image features a basophil.

Figure 10. One-shot learning with the user and assistant roles using a basic prompt.

Finally, we introduced the system role, alongside the user and the assistant, to instruct the model to act as a "pathologist who specializes in the analysis of peripheral blood smears". Moreover, we also tried adding to the system message a description of the morphological characteristics of each leukocyte type, the same

ones used to perform few-shot CoT, to provide more context and improve model performance.

3.4. GPT API:

To evaluate GPT4's performance in blood cell image analysis and classification, we predominantly used the vision model GPT-4V (gpt-4-vision-preview). Since this model is not available in GPT's playground, we worked with OpenAI's API, an application programming interface that allowed us to interact with the desired model for chat completion. The code was written in Python, although we used the JSON format to create the prompt and obtain structured output responses from GPT. The Google Colab platform was employed to write and execute the code. In Figure 11, we show the code used to perform zero-shot CoT, which contains the basic structure utilised for all other prompt strategies with GPT-4. The images used as input had to be converted into a base64 string before using them in the prompt.

```
import base64
import requests

# OpenAI API Key
api_key = "INSERT_YOUR_OPENAI_KEY_HERE"

# Function to encode the image
def encode_image(image_path):
 with open(image_path, "rb") as image_file:
  return base64.b64encode(image_file.read()).decode('utf-8')

# Path to your image
image_path = "/content/SNE_968729.jpg"

# Getting the base64 string
base64_image = encode_image(image_path)

headers = {
 "Content-Type": "application/json",
 "Authorization": f"Bearer {api_key}"
}

payload = {
 "model": "gpt-4-vision-preview",
 "messages": [
   {"role": "user",
    "content": [
     {
      "type": "text",
```

```
        "text": "Identify the blood cell type in this image. Provide a step-by-step reasoning
before providing your answer:"
      },
      {
        "type": "image_url",
        "image_url": {
          "url": f"{base64_image}"
        }
      }
    ]
  }
 ],
 "max_tokens": 2000,
 "temperature": 0,
 "top_p": 0
}

response = requests.post("https://api.openai.com/v1/chat/completions",
headers=headers, json=payload)

print(response.json())
```

Figure 11. Code used to perform zero-shot CoT with GPT-4V.

Due to the recent announcement that OpenAI is deprecating the GPT-4V model on December 6, 2024, we also tested some of the best-performing prompt strategies with two other GPT-4 models with vision capabilities: the newly released GPT-4 Omni model (gpt-4o) and GPT-4 Turbo (gpt-4-turbo).

3.5. Claude Workbench:

To study the capabilities of the Claude3 models Haiku, Sonnet, and Opus, we used the playground provided by Anthropic. Using this console, we were able to easily upload up to 20 images and play with the user and assistant roles. To perform few-shot learning, we adapted the nomenclature shown in the bibliography and guides provided by Anthropic (Fig. 12).

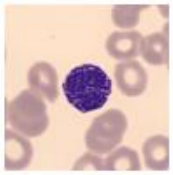
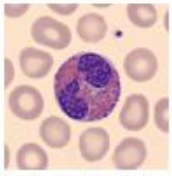
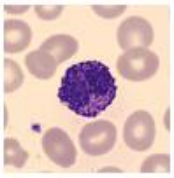
Basic Prompt	
USER:	<example> Text: Identify the blood cell type in this image.  Output: This image features a basophil. </example> <example> Text: Identify the blood cell type in this image.  Output: This image features an eosinophil. </example> (...) Text: Identify the blood cell type in this image.  Output:
ASSISTANT OUPUT:	This image features a basophil.

Figure 12. One-shot learning with Claude3 using the Anthropic playground.

3.6. LLaVa Setup:

To study the performance of LLaVa's latest version (llava-v1.6-34b) we first started working with its online interface to carry out zero-shot learning. However, this interface is not suited to perform few-shot learning. For this purpose, we used an API, employing Ollama as a framework to run the model and the Jarvislabs platform to rent a powerful and affordable GPU. Moreover, we had to adapt the prompting strategies and the code used for few-shot learning to this new model. An example of this code, to perform one-shot with both the user and assistant roles, is shown in Figure 13.

```
!pip install ollama

import base64
import requests
import json

# Function to encode the image
def encode_image(image_path):
 with open(image_path, "rb") as image_file:
  encoded_string = base64.b64encode(image_file.read())
  return encoded_string.decode('utf-8')

# Path to your image
image_path = "/content/BNE_752441.jpg"

# Getting the base64 string
base64_image = encode_image(image_path)
basophil_image = encode_image("/content/BA_229935.jpg")
eosinophil_image = encode_image("/content/EO_74387.jpg")
lymphocyte_image = encode_image("/content/LY_164944.jpg")
monocyte_image = encode_image("/content/MO_85774.jpg")
neutrophil_image = encode_image("/content/SNE_746001.jpg")
band_image = encode_image("/content/BNE_53949.jpg")

import ollama
from ollama import Client
client = Client(host='https://fca4526b7b161.notebooksc.jarvislabs.net/')
response = client.chat(model='llava:34b-v1.6',
                options = {"temperature": 0},
                messages=[
  {
   "role": "user",
   "content": "Identify the blood cell type in this image:",
   "image": f"{basophil_image}"
  },
  {
   "role": "assistant",
   "content": "This image features a basophil."
  },
  {
   "role": "user",
   "content": "Identify the blood cell type in this image:",
   "image": f"{eosinophil_image}"
  },
  {
   "role": "assistant",
   "content": "This image features an eosinophil."
  },
      {
```

```
    "role": "user",
    "content": "Identify the blood cell type in this image:",
    "image": f"{lymphocyte_image1}"
  },
  {
    "role": "assistant",
    "content": "This image features a lymphocyte.”
  },
  {
    "role": "user",
    "content": "Identify the blood cell type in this image:",
    "image": f"{monocyte_image}"
  },
  {
    "role": "assistant",
    "content": "This image features a monocyte."
  },
  {
    "role": "user",
    "content": "Identify the blood cell type in this image:",
    "image": f"{neutrophil_image}"
  },
  {
    "role": "assistant",
    "content": "This image features a neutrophil."
  },
  {
    "role": "user",
    "content": "Identify the blood cell type in this image:",
    "image": f"{band_image}"
  },
  {
    "role": "assistant",
    "content": "This image features a band neutrophil.”
  },
  {
    "role": "user",
    "content": "Identify the blood cell type in this image:",
    "image": f"{base64_image}"
  }
])

print(response['message']['content'])
```

Figure 13. Code used to perform one-shot learning with llava-v1.6-34b, employing both the user and assistant roles.

4. Results and Discussion

4.1. Morphological description of peripheral blood cells:

Peripheral blood contains red blood cells (erythrocytes), white blood cells (leukocytes) and platelets, but also immature forms of some of these cells. Erythrocytes are non-nucleated, bi-concave small cells, about 7-8um in diameter, mostly responsible for the transport of gases and oxygen throughout the body. Nucleated erythroid precursors, also called erythroblasts, can also be found in peripheral blood, although an abundance of this type of cells can be associated with severe anaemia. Platelets, which are also non-nucleated, have an important role in blood clotting. They are smaller than erythrocytes, with a diameter of 2-4um. Larger platelets or macrothrombocytes can also be detected in blood films as a sign of immune thrombocytopenia or other disorders.

Leukocytes are nucleated cells that play a role in protecting the body against infection and foreign bodies. They can be classified as granulocytes or agranulocytes. Basophils, neutrophils, and eosinophils are granulocytes, which contain specific granules along their cytoplasm with enzymes released upon infection or other immune responses, such as allergic reactions. On the other hand, monocytes and lymphocytes are agranulocytes. This means they lack specific granules in their cytoplasm, although they can contain azurophilic (lysosome) granules. However, their appearance and morphology can be easily distinguished when stained with Giemsa or Leishman. After extensive bibliographic research, we have created Table 5, which shows the main morphological differences used by clinicians to identify these leukocytes in peripheral blood films.

Table 5. Morphological characteristics of neutrophils, band neutrophils, eosinophils, basophils, lymphocytes, and monocytes.

BLOOD CELLS	*SIZE*	*NULEOCYTOPLASMIC RATIO*	*NUCLEUS SHAPE*	*SEGMENTS OF THE NUCLEUS*
NEUTROPHIL	intermediate	low	segmented	2 to 5
BAND NEUTROPHIL	intermediate	low	band	0
EOSINOPHIL	intermediate	low	segmented	2
BASOPHIL	intermediate	low	segmented	variable
LYMPHOCYTE	small	high	round	mononuclear
MONOCYTE	high	moderate	indented	mononuclear

BLOOD CELLS	*NUCLEAR CHROMATIN*	*CYTOPLASM*	*COLOUR CYTOPLASM*	*GRANULATION*
NEUTROPHIL	condensed/heterogeneous	wide	azurophilic	azurophil
BAND NEUTROPHIL	condensed/heterogeneous	wide	azurophilic	azurophil
EOSINOPHIL	condensed/heterogeneous	wide	eosinophilic	round eosinophilic
BASOPHIL	condensed/heterogeneous	wde	azurophilic	round basophilic
LYMPHOCYTE	condensed/heterogeneous	scant	basophilic	occasional
MONOCYTE	low condensed	moderate	grayish	in fine sand

As mentioned in the methods section, we fed these tables to ChatGPT to generate descriptions for each leukocyte type. These are the descriptions obtained, after reviewing them:

- Basophils: Basophils have an intermediate size, low nucleocytoplasmic ratio, and a segmented nucleus whose segments can vary. They contain dark, condensed/heterogeneous nuclear chromatin and a wide azurophilic cytoplasm with round basophilic granulation. The large number of dark purplish granules often make the nucleus difficult to see.

- Eosinophils: Eosinophils, with an intermediate size and low nucleocytoplasmic ratio, are recognized by their distinctly segmented nucleus, with 2 segments, and condensed/heterogeneous nuclear chromatin. Their wide cytoplasm is eosinophilic, complemented by round dark-pink granules.

- Lymphocytes: Lymphocytes are small but have a high nucleocytoplasmic ratio. Their nucleus is round and mononuclear, with condensed/heterogeneous chromatin without nucleoli. Their scant, basophilic cytoplasm shows only occasional granulation.

- Monocytes: Monocytes are characterized by their high size and moderate nucleocytoplasmic ratio. They exhibit an indented nucleus described as mononuclear, with low condensed nuclear chromatin. The cytoplasm of a monocyte is moderate in volume, displaying a grayish colour that accompanies its fine sand-like granulation.

- Neutrophils: Neutrophils are intermediate-sized blood cells characterized by a low nucleocytoplasmic ratio. Its nucleus is segmented into 2 to 5 parts and features condensed/heterogeneous chromatin with no nucleoli present. The cytoplasm is wide and azurophilic, containing azurophil granulation.

- Band neutrophils: Band neutrophils are a less mature form of a neutrophil. This form has an intermediate size and low nucleocytoplasmic ratio, but its nucleus takes a distinct band shape with no segments. Featuring similar nuclear

chromatin and absence of nucleoli, its azurophilic cytoplasm also houses azurophil granulation.

These morphological descriptions were subsequently used to carry out few-shot prompting with the selected LVLMs and perform in-context learning.

4.2. GPT-4 Performance:

4.2.1. GPT-4V:

Firstly, we performed zero-shot learning, as described in the methods section, to evaluate the capabilities of this fundamental model. We tried different strategies, including a basic prompt, role prompting, and zero-shot CoT. The accuracy rates obtained by using these prompts on 5 random images for each cell type were 25.71%, 22.86% and 25.71%, respectively. However, we could observe that the model did not identify correctly any platelets or erythroblasts, instead classifying them mostly as neutrophils and lymphocytes. Thus, considering only the results for the leukocyte images, the accuracy improved to 36%, 32% and 36%. Apart from the identification of the cell type, the model also provided us with descriptions of the cells in the image with varying degrees of concretion and specificity. With a basic prompt, descriptions were more generic and broader, while by using role prompting they were usually a bit more precise, pointing out more morphological characteristics of the cells. In the case of zero-shot CoT, descriptions were

Apart from the identification of the cell type, the model also provided us with descriptions of the cells in the image with varying degrees of concretion and specificity. With a basic prompt, the descriptions were more generic and broader, while by using role prompting they were usually a bit more precise, pointing out more morphological characteristics of the cells. In the case of zero-shot CoT, the descriptions were much more specific, providing detailed information about the cell size, the nucleus and the cytoplasm before identifying the cell type, as we instructed (Figure 14).

Basic Prompt

The cell in the image appears to be a white blood cell, specifically a neutrophil. Neutrophils are a type of granulocyte, characterized by their multi-lobed nucleus and granular cytoplasm when stained and viewed under a microscope. They are part of the body's immune system and play a key role in defending against infections by ingesting bacteria and other pathogens.

Role Prompting

This is an image of a peripheral blood smear showing a white blood cell, specifically a neutrophil. Neutrophils are a type of granulocyte, characterized by their multi-lobed nucleus, which usually consists of 2-5 lobes connected by thin strands of chromatin, and their cytoplasm contains fine granules. Neutrophils are part of the body's innate immune system and are the most abundant type of white blood cells in most mammals. They play a key role in defending against infection by ingesting bacteria, dead cells, and other debris through the process of phagocytosis.

Zero-Shot CoT

To identify the type of blood cell in the image, we can follow these steps:

1. **Cell Size**: Compare the size of the cell in question to the surrounding red blood cells (RBCs). The cell in question appears to be larger than the surrounding RBCs, which is typical for white blood cells (WBCs).
2. **Nucleus**: Look at the nucleus of the cell. The cell in the image has a lobed nucleus, which is a characteristic of certain types of WBCs.
3. **Cytoplasm**: Examine the cytoplasm. The cytoplasm appears to be a light blue color, which is typical for WBCs, and there are no granules visible that are characteristic of granulocytes.
4. **Nucleus Shape**: The nucleus has a distinctive shape with two lobes connected by a thin strand, which is a key feature of a specific type of WBC.

Based on these observations, the cell in the image is most likely a neutrophil, which is a type of granulocyte. Neutrophils are characterized by their multi-lobed nucleus, which usually consists of 2-5 lobes connected by thin strands, and their role in the immune system is to engulf and destroy pathogens. The absence of visible granules in this image could be due to the resolution or the staining technique, but the lobed nucleus is a strong indicator of a neutrophil.

Figure 14. Outputs of GPT-4V obtained for the same image when using a basic prompt, role prompting or zero-shot CoT.

Next, we performed one-shot and two-shot with GPT-4V, only with the leukocyte images (basophils, eosinophils, lymphocytes, monocytes and neutrophils), using only a user message or by intercalating the user and assistant roles. Although we also wanted to test three-shot learning with this model, we could not carry out this technique with GPT-4V, as prompts are restricted to a maximum of 10,000 tokens. Alternatively, to try to improve model performance and carry out an approximation to three-shot learning while maintaining the same number of images, we created a hybrid prompt with 3 sample images for lymphocytes and monocytes, which were the least recognized when using two-shot learning, and only 1 image for basophils and eosinophils. The accuracy rates are presented in Table 6.

Table 6. Accuracy rate of GPT-4V for one-shot, two-shot, and an approximation to three-shot learning, using 3 samples images of lymphocytes and monocytes and only one for basophils and eosinophils.

		ONLY USER	WITH ASSISTANT
ONE-SHOT	Basic Prompt	60.00	44.00
	CoT	48.00	52.00
	CoT (Step-by-Step)	68.00	72.00
TWO-SHOT	Basic Prompt	72.00	72.00
	CoT	68.00	68.00
	CoT (Step-by-Step)	56.00	68.00
THREE-SHOT ADAPTED	Basic Prompt	64.00	-

The best results were obtained when performing two-shot learning with a basic prompt (without intermediate reasoning steps), using only the user role or with the assistant and user roles combined. Surprisingly, we did not see an improvement in the accuracy when carrying out CoT, although this strategy usually performs better for complex tasks. On the other hand, the three-shot learning approach did not improve model accuracy. It recognised correctly fewer eosinophils and basophils

and, despite using 3 sample images for lymphocytes and monocytes, no increase in their identification was observed.

With regards to cell description, we could test out how the examples and intermediate reasoned steps shaped model behaviour. When using a basic prompt, GPT-4V did not deliver a cell description, copying the style of the answer we had provided in the examples. Nonetheless, we obtained more detailed morphological descriptions when performing CoT, which included some of the features we had pointed out in the intermediate reasoned steps. However, more detailed descriptions were obtained by asking for step-by-step reasoning, resulting in a more detailed exploration of the features of the cells, including cell size, nucleus shape, nucleocytoplasmic ratio, cytoplasm colour and granulation (Figure 15).

Basic Prompt

This image features a basophil.

CoT

The blood cell type in this image is a basophil. Basophils are characterized by their large, dark purplish granules that can obscure the nucleus, which is segmented and has condensed/heterogeneous nuclear chromatin. The cytoplasm is wide and azurophilic with round basophilic granulation.

CoT (Step-by-step)

To identify the blood cell type in the provided image, we will follow a step-by-step reasoning process:

1. **Size and Shape**: The cell appears to be of intermediate size compared to the surrounding red blood cells, which is typical for most white blood cells.
2. **Nucleus**: The nucleus is not clearly segmented, which can help differentiate between certain types of white blood cells. It appears to be a single, compact structure, which suggests that it is not a neutrophil (which typically has a multi-lobed nucleus).
3. **Cytoplasm and Granules**: The cytoplasm is filled with dark, coarse granules that obscure the view of the nucleus. This is a key characteristic of certain types of granulocytes.
4. **Color of Granules**: The granules are dark purple or black, which is indicative of basophilic granulation.

Based on these observations, the blood cell in the image is most likely a basophil. Basophils are characterized by their large, darkly-stained granules that can obscure the nucleus and are typically seen in low numbers in a blood smear.

Figure 15. Outputs of the same image obtained by performing few-shot prompting on GPT-4V using a basic prompt, CoT or by asking for a step-by-step reasoning.

As another attempt to improve the model accuracy, we tried implementing the system role, as explained in the methods section. We performed zero-shot learning, with both a role prompting approach and by introducing the leukocyte descriptions in the system message. Performing role prompting using the system did not improve the accuracy of the model (36%), in comparison to the previous zero-shot learning results. However, introducing the cell descriptions in the system message did result in a much better accuracy (56%) obtaining even better performance than with zero-shot CoT. We also tried combining this last strategy with two-shot learning while using the assistant and the user roles. However, we saw a decrease in model accuracy (60%), contrary to what we expected. Considering that most of the wrong answers were identified as band neutrophils, we repeated this process excluding the band neutrophil description from the system message, obtaining an improved accuracy rate of 84% (Table 7).

Table 7. Accuracy rates for one-shot and two-shot learning with the assistant and system roles.

		Role Prompting + Cell Description	
	Role Prompting	With Band Neutrophil	Without Band Neutrophil
ZERO-SHOT	36.00	56.00	-
TWO-SHOT	-	60.00	84.00

So far, we have tried these prompting strategies using a limited number of peripheral blood cell images, only 25. To obtain more robust results, we have tested again the two-shot learning strategies that resulted in the best accuracy rates on 100 new peripheral blood cell images selected at random.

Table 8. Accuracy rate for every leukocyte type using a two-shot strategy using only the user role, intercalating the user and assistant roles, or by incorporating a system message with the description of the cells.

	Two-Shot with a Basic Prompt		
	ONLY USER	**WITH ASSISTANT**	**WITH ASSISTANT AND SYSTEM**
Basophils	65.00	90.00	80.00
Eosinophils	65.00	75.00	90.00
Lymphocytes	45.00	35.00	55.00
Monocytes	25.00	35.00	45.00
Neutrophils	100.00	100.00	100.00
Total accuracy	60.00	67.00	74.00

As we can see in Table 8, the accuracy obtained by testing these two-shot prompting strategies in a bigger pool of images decreases a bit. Nonetheless, the overall accuracy rate when using the assistant role (67%) is slightly better than the one obtained using only the user role (60%). Moreover, by incorporating the assistant role, we accomplished a better accuracy for all leukocyte types except for lymphocytes, which decreased from an accuracy of 45% to 35%. However, the best overall accuracy was accomplished by incorporating the system role (74%), asking for the model to act as a pathologist, and integrating the morphological descriptions for each leukocyte type into the prompt, excluding band neutrophils.

4.2.2. GPT-4 turbo and GPT-4o:

To evaluate the vision capabilities of the new GPT-4 Turbo and GPT-4o models, we tested them with some of the strategies described in the previous section. As a first approach, we performed zero-shot learning using a basic prompt, role prompting or CoT. Thereafter, we carried out two-shot learning with a basic prompt using only the user message, intercalating the user and assistant roles, or including the system role into the prompt (Table 9).

Table 9. Accuracy rates obtained when performing zero-shot and two-shot learning, using a basic prompt, with GPT-4 Turbo and GPT-4o.

		GPT-4 Turbo	GPT-4o
ZERO-SHOT	Basic Prompt	36.00	76.00
	Role Prompting	-	76.00
	CoT	-	64.00
TWO-SHOT (Basic Prompt)	Only User	68.00	84.00
	With Assistant	64.00	84.00
	With Assistant and System	84.00	80.00

When using GPT-4 turbo, we do not observe any improvement in model performance compared to the GPT4V model. However, we see once again an increase in the accuracy rate with two-shot learning when the system role was implemented. On the contrary, we see a substantial improvement in model accuracy with GPT4o. We even get better results without any soft training (76%) than with GPT-4V using two-shot learning (72%). In fact, by using this new model, we could improve the accuracy obtained by two-shot learning on both the user and assistant roles from 72% to 84%. On the other hand, we obtained similar outputs to those of GPT-4V for zero-shot prompting, obtaining more detailed morphological descriptions with the zero-shot CoT approach. In the case of two-shot learning, as we were using a basic prompt without intermediate reasoned steps, we did not obtain cell descriptions.

Finally, considering the enhanced performance of this new GPT4o model, we tried again zero-shot and two-shot learning with the user and assistant roles on 100 images. This way, we could better evaluate the capabilities of this model (Table 10).

Table 10. Accuracy rates of GPT-4o for each leukocyte type using a zero-shot or a two-shot learning approach, intercalating the user and assistant roles, on 100 images.

	ZERO-SHOT	TWO-SHOT WITH ASSISTANT
Basophils	30.00	95.00
Eosinophils	55.00	95.00
Lymphocytes	75.00	40.00
Monocytes	70.00	70.00
Neutrophils	70.00	90.00
Total accuracy	60.00	78.00

Once again, we can see that even without any soft training, this model shows a similar performance to the one obtained by GPT4V with two-shot learning. Moreover, by performing two-shot learning with GPT4o, we have been able to increase the overall accuracy from 67% to 78%, without even using the system role. This accuracy is even slightly higher than the one obtained by testing the GPT4V model with 100 images when the system role was integrated into the prompt,

increasing the accuracy from 74% to 78%. Nonetheless, this new model still struggles to recognize lymphocytes, showing an accuracy rate for this leukocyte type of 40%.

4.3. Claude3 Performance:

As a first approach to evaluating Claude3, we performed zero-shot with Opus, its most powerful model, using 5 images for each cell type, including platelets and erythroblasts. The accuracy rates when using a basic prompt, role prompting, or zero-shot CoT were 14.29%, 20% and 5.71%, respectively. However, we could observe again that the model struggles in detecting erythroblasts and platelets, identifying them as eosinophils or neutrophils for the most part. Therefore, by just taking into account the leukocyte images, we obtained better accuracy rates (Table 11).

Table 11. Accuracy rates of Haiku, Sonnet, and Opus with zero-shot learning trying only the leukocyte images

	ZERO-SHOT		
	Basic Prompt	**Role Prompting**	**CoT**
Haiku	8.00	16.00	28.00
Sonnet	20.00	20.00	20.00
Opus	20.00	24.00	4.00

At first glance, we can observe that role prompting slightly increased the accuracy of the Haiku and Opus models, from 8% to 16% and from 20% to 24%, respectively. Moreover, we can see an improvement in the accuracy of the Haiku model when CoT was performed, showing the best performance for all models and prompting techniques (28%). Interestingly, some of the answers provided by Claude3, while identifying the red blood cells surrounding our cell of interest, did not recognise the latter. They are rather mentioned as an artefact or abnormality in most cases. This type of output was especially common when performing CoT with Opus, which is why the accuracy rate is 4%.

Once again, we performed one-shot learning by containing the prompt in a single user message or incorporating the assistant role into the prompt (Table 12).

Table 12. Accuracy rates of Haiku, Sonnet and Opus with one-shot learning using a basic prompt, CoT or CoT (Step-by-step). Columns show the results with or without the assistant role.

		ONE-SHOT	
		ONLY USER	WITH ASSISTANT
Basic	Haiku	4.00	0.00
	Sonnet	32.00	32.00
	Opus	20.00	8.00
CoT	Haiku	0.00	0.00
	Sonnet	48.00	0.00
	Opus	4.00	0.00
CoT (Step-by-Step)	Haiku	20.00	24.00
	Sonnet	40.00	36.00
	Opus	32.00	44.00

Surprisingly, the best performing model was Sonnet, which did surpass Haiku and Opus in accuracy rate. Accordingly, the best accuracy (48%) was obtained with Sonnet by performing Chain of Thought in a single user prompt. Alternatively, Haiku showed poor performance, as most cells were identified as platelets when using a basic prompt or CoT. On the other hand, incorporating the assistant role into the prompt did not improve the overall accuracy of the Claude3 models.

Moreover, we did also carry out two-shot and three-shot learning with Sonnet and Opus, using only the user role. Haiku was excluded due to its poor results. Surprisingly, we did not observe a significant improvement in model performance compared to the one-shot learning results. The only significant improvement was obtained by using the basic prompt with Sonnet, which increased the accuracy rate from 32% to 56%. Moreover, we obtained the same accuracy when performing two-shot and three-shot learning (Figure 13).

Table 13. Accuracy rates of Sonnet and Opus with two-shot or three-shot learning using only the user role.

		TWO-SHOT	THREE-SHOT
Basic Prompt	Sonnet	56.00	56.00
	Opus	24.00	24.00
CoT	Sonnet	48.00	-
	Opus	28.00	-
CoT (Step-by-step)	Sonnet	32.00	-
	Opus	20.00	-

Finally, we tried Sonnet again with the best performing strategy (two-shot learning with a basic prompt) on 100 new images. This resulted in an underwhelming overall accuracy of 47%. Moreover, the model struggled to identify eosinophils and lymphocytes (Figure 14).

Table 14. Accuracy rate of Sonnet for each leukocyte type when performing two-shot learning with a basic prompt on 100 images. Only the user role was used.

	TWO-SHOT (Basic Prompt)
Basophils	80.00
Eosinophils	0.00
Lymphocytes	10.00
Monocytes	65.00
Neutrophils	80.00
Total accuracy	47.00

4.4. LLaVa Performance:

To evaluate the performance of this last LVLM, we tried once again zero-shot, one-shot and two-shot learning. Starting with zero-shot learning, we did observe similar results to Claude3, both by using all cell types (including platelets and erythroblasts) or only leukocytes (Table 15).

Table 15. Accuracy rates of LLaVa 1.6 using zero-shot learning with a basic prompt, role prompting or CoT. Columns show the results obtained with all the cell types or only with leukocyte images.

		ALL CELLS	ONLY LEUKOCYTES
ZERO-SHOT	Basic Prompt	11.43	12.00
	Role Prompting	14.29	20.00
	CoT	0.00	0.00

When we carried out zero-shot CoT, most cells were identified as red blood cells, hence the 0% accuracy. On the other hand, when we used a basic prompt or role prompting, most images were recognised as lymphocytes, although some outputs were just classified as leukocytes or were not specified.

Afterwards, we performed one-shot learning using only the user role in a single prompt, the same as we did with GPT4-V and Claude3. However, this prompt provided us with the same inconclusive result for all images:

"The image you've provided is too small and blurry for me to accurately identify the specific type of blood cell. If you can provide a clearer, higher-resolution image or more information about the cell, I may be able to help you identify it."

Therefore, we carried out one-shot with the assistant role, following the examples provided by the bibliography. By using this approach, we were able to obtain proper responses. We also performed two-shot learning with this same strategy (Table 16). Nonetheless, these prompting techniques did not improve model performance. The highest accuracy we were able to achieve was 20%, and we could only recognise neutrophils or lymphocytes correctly when performing two-shot or one-shot learning, respectively. On the other hand, when using a basic prompt, the images were recognised either as red blood cells or platelets, thus obtaining an accuracy rate of 0%.

Table 16. Accuracy rate of LLaVa 1.6 with one-shot and two-shot learning using a basic prompt, CoT or CoT (Step-by-step). The assistant role was incorporated into the prompt.

	ONE-SHOT	TWO-SHOT
Basic	0.00	0.00
CoT	0.00	20.00
CoT (Step-by-Step)	20.00	20.00

Considering the unsatisfactory performance of LlaVa, we did not continue working with this model.

4.5. Comparison between models and prompting techniques:

After reviewing the accuracy rates of all prompting techniques for each model, we would like to provide a more comprehensive comparison between them. Firstly, we have put together Table 17, with all accuracy rates obtained for each model and prompting strategy.

Table 17. Comparison of all the accuracy rates between the LVLMs and prompting techniques. The highest values are shown in green.

			GPT-4			Claude3			LLaVa 1.6
			GPT-4V	GPT-4 turbo	GPT-4o	Haiku	Sonnet	Opus	
ZERO-SHOT	All cells	Basic Prompt	25.71					14.29	11.43
		Role Prompting	22.86					20.00	14.29
		CoT	25.71					5.71	0.00
	Only leukocytes	Basic Prompt	36.00	36.00	76.00	8.00	20.00	20.00	12.00
		Role Prompting	32.00		76.00	16.00	20.00	24.00	20.00
		CoT	36.00		64.00	28.00	20.00	4.00	0.00
ONE-SHOT	Only user	Basic Prompt	60.00			4.00	32.00	20.00	
		CoT	48.00			0.00	48.00	4.00	
		CoT (Step-by-step)	68.00			20.00	40.00	32.00	
	With assistant	Basic Prompt	44.00			0.00	32.00	8.00	0.00
		CoT	52.00			0.00	0.00	0.00	0.00
		CoT (Step-by-step)	72.00			24.00	36.00	44.00	20.00
	With assistant and system	Role Prompting	36.00						
		Cell description	56.00						
TWO-SHOT	Only user	Basic Prompt	72.00	68.00	84.00				
		CoT	68.00						
		CoT (Step-by-step)	56.00						
	With assistant	Basic Prompt	72.00	64.00	84.00		56.00	24.00	0.00
		CoT	68.00				48.00	28.00	20.00
		CoT (Step-by-step)	68.00				32.00	20.00	20.00
	With assistant and system	Cell description	84.00	84.00	80.00				
THREE-SHOT		Basic Prompt	64.00				56.00	24.00	

Moreover, we have created barplots to represent some of the most notable results obtained for zero-shot, one-shot and two-shot learning, comparing the accuracy rates between models and prompts.

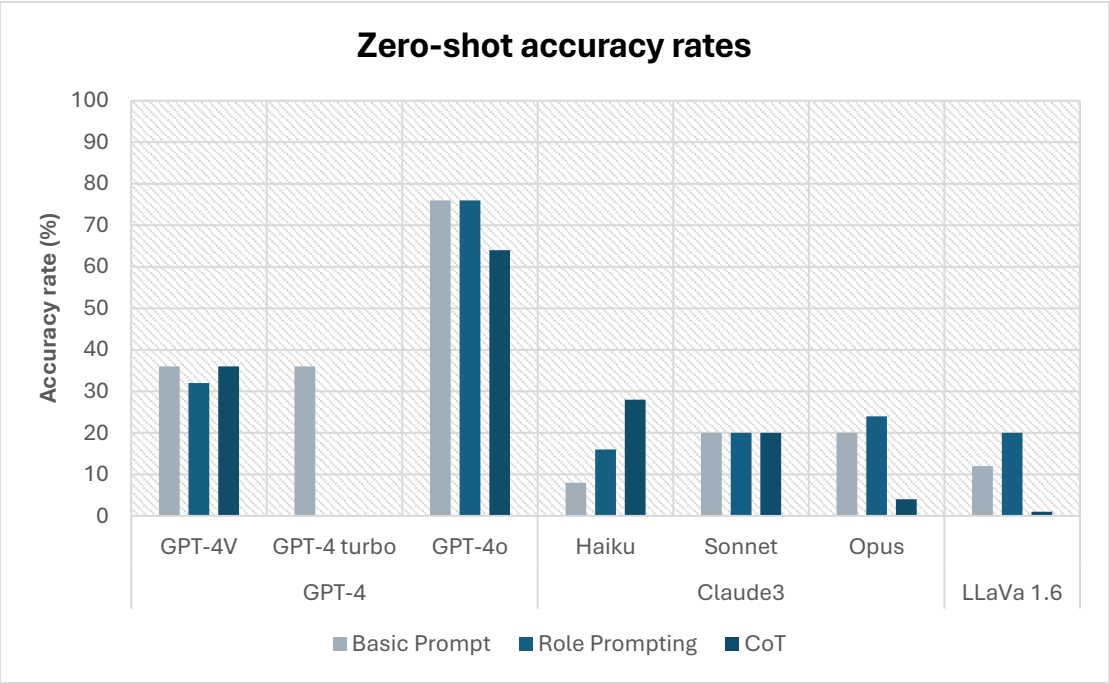


Figure 16. Barplot of the zero-shot learning accuracy rates, using a basic prompt, role prompting, or CoT with the GPT-4, Claude and LLava 1.6 models.

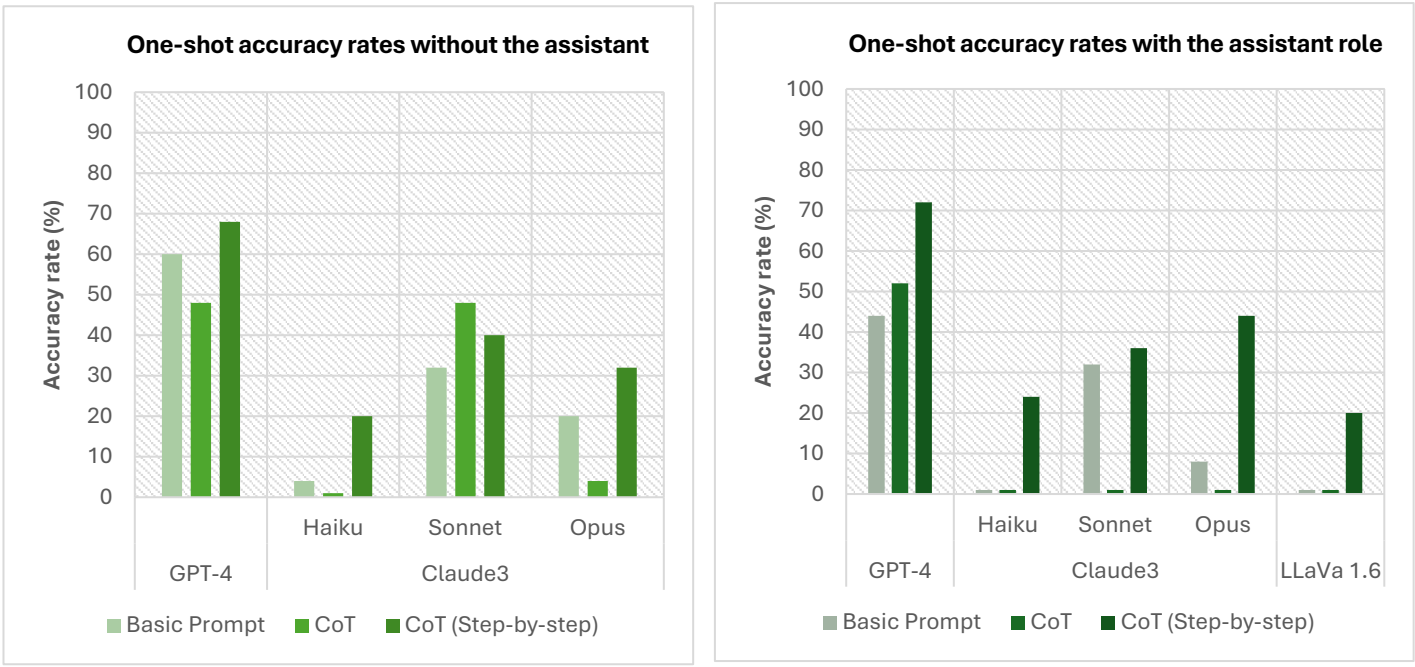


Figure 17. Barplot of the one-shot learning accuracy rates, using a basic prompt, CoT or CoT (Step-by-step) and the GPT-4, Claude3 and LLaVa models. Results are shown when intercalating the assistant and user roles (right) or by using only the user role (left).

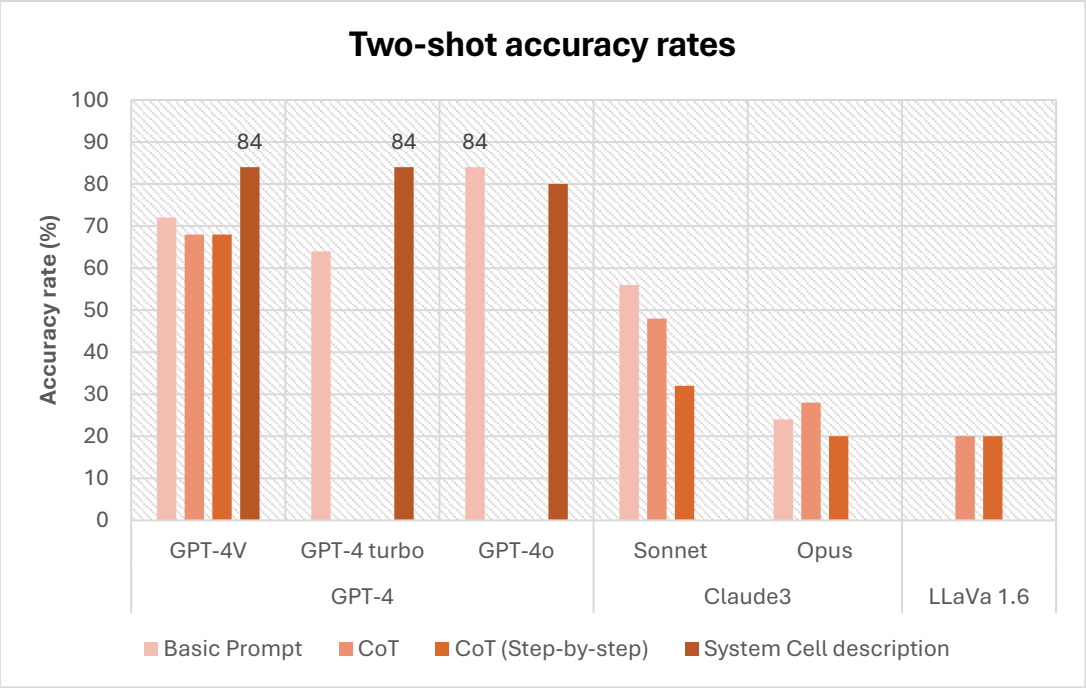


Figure 18. Barplot of the two-shot learning accuracy rates, using a basic prompt, CoT or CoT (Step-by-step) with the assistant and user roles, or by incorporating morphological cell descriptions into the system message.

First, we can observe that GPT-4 is the best-performing model in all instances. Moreover, the GPT-4o model provided twice the accuracy of GPT-4V when performing zero-shot learning, showing proficiency in VQA. Although we can see an overall improvement when carrying out one-shot prompting, in contrast to zero-shot learning, the best performance was obtained with two-shot prompting. Providing the model with two examples of each cell type improved the accuracy by up to 84%. Overall, the best strategies and models were GPT-4V, GPT-4 turbo, and GPT-4o, incorporating the assistant and system roles while using a basic prompt without intermediate reasoned steps. Nonetheless, as we described in previous sections, when trying these strategies in a larger number of images, we could obtain more reliable accuracy rates. Thus, we could determine that the best-performing method and LVLM combination, with an overall accuracy rate of 78%, is GPT-4o using a two-shot learning strategy, incorporating the assistant role but without adding intermediate reasoning steps.

5. Web application

With the objective of portraying these results in a more practical and tangible way, as well as providing researchers with an easy tool to analyse and classify white blood cells, we created a web application.

For this purpose, we used Flask as a web application framework and created a Python script by taking advantage of the GPT API. We used the GPT4o model with two-shot learning and with the assistant role, which provided the best overall accuracy when tested on 100 images (78%). Moreover, we generated a basic HTML form to collect user input and give shape to the web application. The full code can be found on the Appendix and GitHub.

In Figure 19, we show an image of this web application, already with an uploaded image and the provided result.

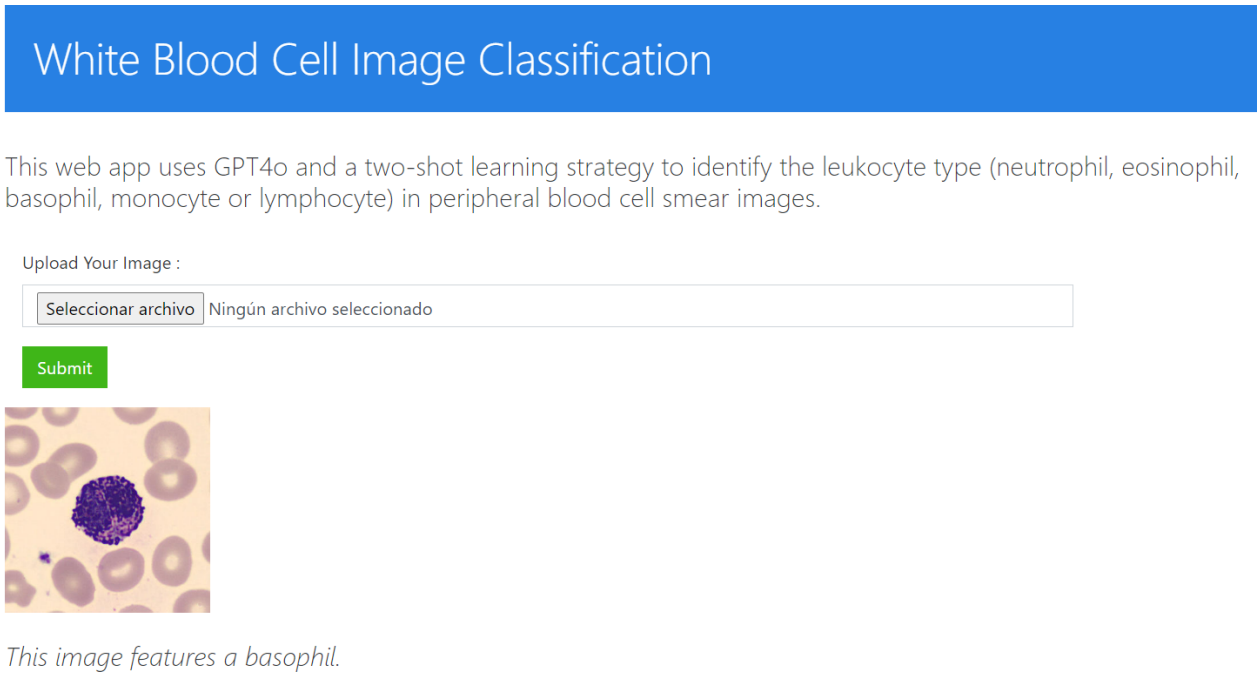


Figure 19. Image of the web application interface. A basophil image was uploaded and recognised by the application.

Although the interface and the accuracy of the current GPT-4o model should be improved, we believe this is a good first step in the application of LVLMs for the analysis of peripheral blood cell images. Moreover, this script works as a template to use with other GPT-4 prompts and models.

6. Conclusions

Firstly, we successfully did bibliographical research on the state-of-the-art of LVLMs and their application for biomedical imaging. We explored the evolution of LVLMs, some of the most advanced large multimodal models, and a few specialised biomedical models. Likewise, we researched different prompt engineering strategies to explore and improve the capabilities of GPT-4, Claude3 and LLava on peripheral blood cell analysis and classification. Moreover, we were able to research the morphological characteristics of these cells, create tables to summarise these distinctive features, and generate specific descriptions for each cell type with the help of ChatGPT. These morphological descriptions were subsequently used to carry out few-shot prompting with the selected LVLMs and perform in-context learning.

Furthermore, we have identified GPT4 Omni as the best-performing method for the identification of peripheral blood cell images. Although its best overall accuracy was obtained when carrying out two-shot learning, its results on zero-shot were also much superior to those of its competitors. This means that, even without any soft training, GPT-4 Omni shows better proficiency in basic medical knowledge. This is particularly convenient as GPT-4 Omni is one of GPT-4's cheapest and fastest models.

On the other hand, LLaVa showed the worst performance of all LVLMs. This model struggled to recognise white blood cells, identifying a large part of them as red blood cells or platelets. Moreover, soft-training the model by one-shot or two-shot learning did not improve accuracy. Regarding Claude3, we recognised Sonnet as the best-performing model, improving its precision when carrying out two-shot prompting. However, the best accuracy rates obtained by Claude3 barely surpassed 50%, showing poor performance in comparison to GPT-4V.

In terms of the inclusion of the assistant role, we obtained mixed results, depending on the model. For GPT-4, the intercalating user and assistant responses did increase by a small margin the accuracy rate. However, we could not detect this improvement with the Claude3 models when working with Anthropic's Workbench. In the case of LLaVa, the use of the assistant role was mandatory for few-shot learning, otherwise, the model would detect the images as "blurry" and did not analyse the cells properly.

On another note, adding more context to the model through role prompting did not improve its capabilities, although the outputs contained more precise and in-depth descriptions of the cells. Moreover, the addition of intermediate reasoning steps by performing chain of thought did not improve either model accuracy. For the most part, using a simpler and straightforward approach results in the best performing strategies. Nonetheless, the inclusion of morphological descriptions for each cell type in the system role did improve the precision of some models. This could be due to the fact that by introducing these descriptions in the system and not as part of the

chain of thought, we are providing a much more comprehensive and less restraining context for the overall conversation between the user and the assistant.

Finally, we were able to determine that the best-performing method and LVLM combination is GPT-4o when using a two-shot learning strategy, with the addition of the assistant role but without the use of intermediate reasoning steps. By using this approach on 100 images of leukocytes, we attained an accuracy rate of 78%. Moreover, by using this strategy, we were able to create a web application capable of classifying white blood cells from peripheral blood cell smears with a 78% accuracy rate.

Although we have accomplished a decent performance by simply exploring prompt engineering techniques in LVLMs, more precise and accurate machine learning techniques have been published, reporting up to a 96.2% classification accuracy for peripheral blood cell images using convolution neural networks (Acevedo et al. 2019). In addition, it is important to point out that the performance displayed by these LVLMs is not reliable enough and, on account of the severity of possible errors in the medical field, they should not be used as a diagnostic tool. Nonetheless, we believe that due to the rapid advancement of large language-vision models and the progressive cheapening of their costs, LVLMs could become a great asset in the analysis of pathology images, working as an assistant for quick blood cell description and classification.

7. Future perspectives

For future work, we intend to try more advanced prompting techniques with GPT-4 and Claude3. Considering the good performance of GPT-4o, we would like to carry out self-consistency, prompt chaining, and tree of thoughts (ToT) with this model. As another way to try to improve model performance, if we manage to overcome token limitations, we want to continue exploring few-shot prompting with GPT-4 using up to 10 examples. Another thing we would like to implement is function calling to obtain more structured data back from the models and to be able to perform multiple calls together.

Furthermore, we plan on exploring other LVLMs for the analysis and classification of peripheral blood cells. Another interesting approach would be to evaluate the capabilities of some specialised models trained on medical data, such as LLava-Med or Med-Flamingo, and compare their performance with that of the foundational models. Moreover, due to the rapid progress in this field, we will need to keep track of emerging models and evaluate their capabilities for medical image analysis.

Once we do a deeper exploration of prompt engineering with these LVLMs, we plan on fine-tuning the best-performing models by retraining them on hundreds of images of peripheral blood cell smears to generate a specialised model for white blood cell classification.

Finally, if the results are favourable, we could try translating this strategy to the analysis of other types of pathological images that have been poorly explored, like stained histology samples or tissue biopsies.

8. Bibliography

Acevedo, A., Alférez, S., Merino, A., Puigví, L., & Rodellar, J. (2019). Recognition of peripheral blood cell images using convolutional neural networks. Computer methods and programs in biomedicine, 180, 105020. https://doi.org/10.1016/j.cmpb.2019.105020

Acevedo, A., Merino, A., Alférez, S., Molina, Á., Boldú, L., & Rodellar, J. (2020). A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. Data in brief, 30, 105474. https://doi.org/10.1016/j.dib.2020.105474

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., . . . Zoph, B. (2023). GPT-4 Technical Report. arXiv:2303.08774

Adewoyin, A. S., & Nwogoh, B. (2014). Peripheral blood film - a review. Annals of Ibadan postgraduate medicine, 12(2), 71–79.

Anthropic. (2024, March 4) Introducing the next generation of claude. Anthropic. https://www.anthropic.com/news/claude-3-family

Baktash, J. A., & Dawodi, M. (2023). Gpt-4: A Review on Advancements and Opportunities in Natural Language Processing. arXiv:2305.03195

Carolan, K., Fennelly, L., & Smeaton, A. F. (2024). A Review of Multi-Modal Large Language and Vision Models. arXiv:2404.01322

Chiruvella, V., & Guddati, A. K. (2021). Ethical Issues in Patient Data Ownership. Interactive journal of medical research, 10(2), e22269. https://doi.org/10.2196/22269

Cui, C., Yang, H., Wang, Y., Zhao, S., Asad, Z., Coburn, L. A., Wilson, K. T., Landman, B. A., & Huo, Y. (2023). Deep multimodal fusion of image and non-image data in disease diagnosis and prognosis: a review. Progress in biomedical engineering (Bristol, England), 5(2), 10.1088/2516-1091/acc2fe. https://doi.org/10.1088/2516-1091/acc2fe

Gao, W., Deng, Z., Niu, Z., Rong, F., Chen, C., Gong, Z., Zhang, W., Xiao, D., Li, F., Cao, Z., Ma, Z., Wei, W., & Ma, L. (2023). OphGLM: Training an Ophthalmology Large Language-and-Vision Assistant based on Instructions and Dialogue. arXiv:2306.12174

Hartsock, I., & Rasool, G. (2024). Vision-Language Models for Medical Report Generation and Visual Question Answering: A Review. arXiv:2403.02469

Jahan, I., Laskar, M. T., Peng, C., & Huang, J. (2023). A Comprehensive Evaluation of Large Language Models on Benchmark Biomedical Text Processing Tasks. arXiv:2310.04270

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., & Iwasawa, Y. (2022). Large Language Models are Zero-Shot Reasoners. arXiv:2205.11916

Li, C., Wong, C., Zhang, S., Usuyama, N., Liu, H., Yang, J., Naumann, T., Poon, H., & Gao, J. (2023). LLaVA-Med: Training a Large Language-and-Vision Assistant for Biomedicine in One Day. arXiv:2306.00890

Li, H., Moon, J. T., Purkayastha, S., Celi, L. A., Trivedi, H., & Gichoya, J. W. (2023). Ethics of large language models in medicine and medical research. The Lancet. Digital health, 5(6), e333–e335. https://doi.org/10.1016/S2589-7500(23)00083-3

Li, P., Yang, J., Islam, M. A., & Ren, S. (2023). Making AI Less "Thirsty": Uncovering and Addressing the Secret Water Footprint of AI Models. arXiv:2304.03271

Liu, H., Li, C., Li, Y., Li, B., Zhang, Y., Shen, S., & Lee, Y. J. (2024, January 30). Next: Improved reasoning, OCR, and world knowledge. LLaVA. https://llava-vl.github.io/blog/2024-01-30-llava-next/

Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2023). Visual Instruction Tuning. arXiv:2304.08485

Moor, M., Huang, Q., Wu, S., Yasunaga, M., Zakka, C., Dalmia, Y., Reis, E. P., Rajpurkar, P., & Leskovec, J. (2023). Med-Flamingo: A Multimodal Medical Few-shot Learner. arXiv:2307.15189

Pellegrini, C., Özsoy, E., Busam, B., Navab, N., & Keicher, M. (2023). RaDialog: A Large Vision-Language Model for Radiology Report Generation and Conversational Assistance. arXiv:2311.18681

Pichai, S., & Hassabis, D. (2023, December 6). Introducing Gemini: Our largest and most capable AI model. Google. https://blog.google/technology/ai/google-gemini-ai/

Qin, Z., Yi, H., Lao, Q., & Li, K. (2022). Medical Image Understanding with Pretrained Vision Language Models: A Comprehensive Study. arXiv:2209.15517

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020

Rillig, M. C., Ågerstrand, M., Bi, M., Gould, K. A., & Sauerland, U. (2023). Risks and Benefits of Large Language Models for the Environment. Environmental science & technology, 57(9), 3464–3466. https://doi.org/10.1021/acs.est.3c01106

Rodellar, J., Alférez, S., Acevedo, A., Molina, A., & Merino, A. (2018). Image processing and machine learning in the morphological analysis of blood cells. International journal of laboratory hematology, 40 Suppl 1, 46–53. https://doi.org/10.1111/ijlh.12818

Shu, C., Chen, B., Liu, F., Fu, Z., Shareghi , E., & Collier, N. (2023). Visual Med-Alpaca: A Parameter-Efficient Biomedical LLM with Visual Capabilities. Visual Med-Alpaca. https://cambridgeltl.github.io/visual-med-alpaca/

Singhal, K., Azizi, S., Tu, T., Mahdavi, S. S., Wei, J., Chung, H. W., Scales, N., Tanwani, A., Cole-Lewis, H., Pfohl, S., Payne, P., Seneviratne, M., Gamble, P., Kelly, C., Babiker, A., Schärli, N., Chowdhery, A., Mansfield, P., Demner-Fushman, D., Agüera Y Arcas, B., … Natarajan, V. (2023). Large language models encode clinical knowledge. Nature, 620(7972), 172–180. https://doi.org/10.1038/s41586-023-06291-2

Singhal, K., Tu, T., Gottweis, J., Sayres, R., Wulczyn, E., Hou, L., Clark, K., Pfohl, S., Neal, D., Schaekermann, M., Wang, A., Amin, M., Lachgar, S., Mansfield, P., Prakash, S., Green, B., Dominowska, E., Arcas, B. A., Tomasev, N., . . . Natarajan, V. (2023). Towards Expert-Level Medical Question Answering with Large Language Models. arXiv:2305.09617

Sorin, V., Barash, Y., Konen, E., & Klang, E. (2023). Large language models for oncological applications. Journal of cancer research and clinical oncology, 149(11), 9505–9508. https://doi.org/10.1007/s00432-023-04824-w

Thawkar, O., Shaker, A., Mullappilly, S. S., Cholakkal, H., Anwer, R. M., Khan, S., Laaksonen, J., & Khan, F. S. (2023). XrayGPT: Chest Radiographs Summarization using Medical Vision-Language Models. arXiv: 2306.07971

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023). LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971

Van, M., Verma, P., & Wu, X. (2024). On Large Visual Language Models for Medical Imaging Analysis: An Empirical Study. arXiv:2402.14162

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need. arXiv:1706.03762

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. arXiv:2201.11903

Yan, Z., Zhang, K., Zhou, R., He, L., Li, X., & Sun, L. (2023). Multimodal ChatGPT for Medical Applications: An Experimental Study of GPT-4V. arXiv:2310.19061

Yan, Z., Zhang, K., Zhou, R., He, L., Li, X., & Sun, L. (2023). Multimodal ChatGPT for Medical Applications: An Experimental Study of GPT-4V. arXiv:2310.19061

Yang, L., Wang, Z., Chen, Z., Liang, X., & Zhou, L. (2023). MedXChat: A Unified Multimodal Large Language Model Framework towards CXRs Understanding and Generation. arXiv:2312.02233

9. Appendix

```
#FLASK WEB APP
from flask import Flask, render_template, request
import base64
import requests

app = Flask(__name__)
api_key = " INSERT_YOUR_OPENAI_KEY_HERE "

def gpt4(base64_image):
  # Function to encode images
  def encode_image(image_path):
    with open(image_path, "rb") as image_file:
      return base64.b64encode(image_file.read()).decode('utf-8')
  # Getting the base64 string
  basophil_image1 = encode_image("BA_229935.jpg")
  basophil_image2 = encode_image("BA_594501.jpg")
  eosinophil_image1 = encode_image("EO_74387.jpg")
  eosinophil_image2 = encode_image("EO_280451.jpg")
  lymphocyte_image1 = encode_image("LY_164944.jpg")
  lymphocyte_image2 = encode_image("LY_320312.jpg")
  monocyte_image1 = encode_image("MO_85774.jpg")
  monocyte_image2 = encode_image("MO_116840.jpg")
  neutrophil_image = encode_image("SNE_746001.jpg")
  band_image = encode_image("BNE_53949.jpg")
  # Generate openai response
  headers = {
    "Content-Type": "application/json",
    "Authorization": f"Bearer {api_key}"
  }
  payload = {
    "model": "gpt-4o",
    "messages": [
      {
        "role": "user",
        "content": [
          {
            "type": "text",
            "text": "Identify the blood cell type in these images:"
          },
          {
            "type": "image_url",
            "image_url": {
              "url": f"{basophil_image1}"
            }
          },
```

```
        {
          "type": "image_url",
          "image_url": {
            "url": f"{basophil_image2}"
          }
        }
      ]
    },
    {
      "role": "assistant",
      "content": [
        {
          "type": "text",
          "text": "These images feature a basophil."
        }
      ]
    },
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Identify the blood cell type in these images:"
        },
        {
          "type": "image_url",
          "image_url": {
            "url": f"{eosinophil_image1}"
          }
        },
        {
          "type": "image_url",
          "image_url": {
            "url": f"{eosinophil_image2}"
          }
        }
      ]
    },
    {
      "role": "assistant",
      "content": [
        {
          "type": "text",
          "text": "These images feature an eosinophil."
        }
      ]
    },
    {
      "role": "user",
```

```
    "content": [
      {
        "type": "text",
        "text": "Identify the blood cell type in these images:"
      },
      {
        "type": "image_url",
        "image_url": {
          "url": f"{lymphocyte_image1}"
        }
      },
      {
        "type": "image_url",
        "image_url": {
          "url": f"{lymphocyte_image2}"
        }
      }
    ]
  },
  {
    "role": "assistant",
    "content": [
      {
        "type": "text",
        "text": "These images feature a lymphocyte."
      }
    ]
  },
  {
    "role": "user",
    "content": [
      {
        "type": "text",
        "text": "Identify the blood cell type in these images:"
      },
      {
        "type": "image_url",
        "image_url": {
          "url": f"{monocyte_image1}"
        }
      },
      {
        "type": "image_url",
        "image_url": {
          "url": f"{monocyte_image2}"
        }
      }
    ]
  },
```

```
    {
      "role": "assistant",
      "content": [
        {
          "type": "text",
          "text": "These images feature a monocyte."
        }
      ]
    },
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Identify the blood cell type in these images:"
        },
        {
          "type": "image_url",
          "image_url": {
            "url": f"{neutrophil_image}"
          }
        },
        {
          "type": "image_url",
          "image_url": {
            "url": f"{band_image}"
          }
        }
      ]
    },
    {
      "role": "assistant",
      "content": [
        {
          "type": "text",
          "text": "These images feature a neutrophil."
        }
      ]
    },
    {
      "role": "user",
      "content": [
        {
          "type": "text",
          "text": "Identify the blood cell type in this image:"
        },
        {
          "type": "image_url",
          "image_url": {
```

```
                    "url": f"{base64_image}"
                }
            }
        ]
      },
    ],
    "max_tokens": 4000,
    "temperature": 0,
    "top_p": 0
  }
  response = requests.post("https://api.openai.com/v1/chat/completions", headers=headers,
json=payload)
  text_response = response.json()
  content_response = text_response['choices'][0]['message']['content']
  return content_response

@app.route('/', methods=['GET', 'POST'])
def index():
  return render_template('index.html')

@app.route("/submit", methods=['GET', 'POST'])
def get_image():
  if request.method == 'POST':
    img = request.files['my_image']
    img_path = "static/" + img.filename
    img.save(img_path)
    with open(img_path, "rb") as image_file:
      image64 = base64.b64encode(image_file.read()).decode('utf-8')
    cell_type = gpt4(image64)
  return render_template("index.html", response=cell_type, img_path=img_path)


if __name__ == '__main__':
  app.run(debug=True)
```

HTML Script:

```
<!DOCTYPE html>
<html lang="en">
<head>
 <title>Leukocyte Image Classification</title>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootswatch@4.5.2/dist/cosmo/bootstrap.min.css">
</head>
<body>
```

```
<div class="container">
 <h1 class="jumbotron bg-primary text-white display-6 p-4">White Blood Cell Image Classification</h1>
 <h4 class="description">This web app uses GPT4o and a two-shot learning strategy to identify the leukocyte type (neutrophil, eosinophil, basophil, monocyte or lymphocyte) in peripheral blood cell smear images.</h4>
 <br>
 <form class="form-horizontal" action="/submit" method="post" enctype="multipart/form-data">

  <div class="form-group">
   <label class="control-label col-sm-2" for="pwd">Upload Your Image :</label>
   <div class="col-sm-10">
    <input type="file" class="form-control" name="my_image" id="pwd">
   </div>
  </div>

  <div class="form-group">
   <div class="col-sm-offset-2 col-sm-10">
    <button type="submit" class="btn btn-success">Submit</button>
   </div>
  </div>
 </form>

 {% if response %}
 <img src="{{img_path}}" height="175px" width="175px">
 <br><br>
  <h4><i> {{response}} </i></h4>

 {% endif %}

</div>

</body>
</html>
```