SmartDermAssist

Rafael Pereira[a], Carla Mendes[a] and Ana Cruz[a]

[a]Computer Science and Communications Research Centre, School of Technology and Management, Polytechnic of Leiria, 2411-901 Leiria, Portugal

ARTICLE INFO

Keywords:
skin cancer detection
LLM
CNN
Conversational agents

ABSTRACT

The skin is a crucial organ that performs essential functions, such as temperature regulation and protection against harmful elements. Skin cancer, with both benign and malignant lesions, is a prevalent and growing global health concern. To enhance the monitoring of skin moles and clarify user's doubts regarding any dermatology topic we introduce the SmartDermAssist System with its cutting-edge features, which include seamless doctor-patient communication, a sophisticated chatbot powered with a Large Language Model (LLM) network for information dissemination, and skin spot analysis leveraging a Convolutional Neural Network (CNN) model.

1. Introduction

The skin plays a vital role in regulating body temperature, protecting internal organs from harmful elements like UV rays and microbes, and enabling sensations of touch, heat, and cold. Comprised of the epidermis, dermis, and hypodermis layers, the skin can develop two types of lesions associated with skin cancer: benign and malignant. In benign lesions, melanin deposits are typically present in the epidermis layer. Conversely, malignant lesions exhibit abnormal and excessive melanin production [1].

Skin cancer is one of the most widespread forms of cancer globally, and its incidence has grown over time. In 2020 alone, an estimated 324,635 individuals were diagnosed with melanoma, the deadliest skin cancer. While it can affect people of all ages, older individuals with a history of sun exposure are among the most common cases. However, it is worth noting that skin cancer can also develop in younger individuals, including those below 30. It is one of the most frequently diagnosed cancers among young adults, particularly women. Approximately 2,400 cases of melanoma were estimated to be diagnosed in individuals aged 15 to 29 in the year 2020 [2].

According to the World Health Organization, there is an estimated growth rate of 16.44% in skin cancer cases between 2020 and 2025. Based on the provided data, the number of diagnosed cases is projected to increase from 1,520,000 in 2020 to 1,770,000 in 2025 [3]. Several significant risk factors contribute to the development of skin cancer, such as fair skin, a history of sunburns, prolonged exposure to the sun, a family history of skin cancer, and a weakened immune system [2].

Conversational Agents (CA), commonly referred to as chatbots, are software programs designed to facilitate natural interactions between humans and machines. These agents receive input from users, typically in the form of speech or text, and respond accordingly to maintain a seamless conversation flow using natural language [4].

Chatbots in the healthcare industry hold promise for enhancing the patient journey and enhancing care standards through streamlining communication, optimizing appointment scheduling, providing timely reminders, aiding in diagnostics, facilitating treatment discussions, and supporting patient education. Dermatologists could greatly benefit from chatbots as they experience less volume of patient messages to handle and interruptions to their workflow while also enabling them to effectively prioritize messages, ensuring that high-risk patients are promptly scheduled for in-person appointments. From the patient's standpoint, chatbots provide valuable advantages such as early screening and diagnosis, discussions about treatment choices, follow-ups after visits, and regular monitoring. Furthermore, patients can communicate with the chatbot to detail symptoms, answer questions to provide information to healthcare professionals, receive diagnosis information and explore side effects or any information regarding their treatment or diagnosis [5].

In this paper, we present the SmartDermAssist designed to enhance the monitoring of skin moles and communication process among skin cancer patients and healthcare professionals. The proposed system incorporates various cutting-edge features, including seamless doctor-patient communication, a sophisticated chatbot powered with a LLM network for information dissemination, and skin spot analysis leveraging a CNN model. Therefore, our system aims to provide patients with an efficient and user-friendly platform that empowers them to take control of their skin health while facilitating improved collaboration with healthcare professionals.

- **Revision of state-of-the-art and existing literature**: provide a comprehensive review of the state of the art and relevant literature in the areas of CA, Computer Vision (CV), and LLM within the context of dermatology. This analysis offers insights into the latest advancements and existing research in these fields.
- **SmartDermAssist System**: integrates real-time skin spot analysis using advanced CV techniques with ef-

rafael.m.pereira@ipleiria.pt (R. Pereira); carla.c.mendes@ipleiria.pt (C. Mendes); ana.v.cruz@ipleiria.pt (A. Cruz)

ORCID(s): 0000-0001-8313-7253 (R. Pereira); 0000-0001-7138-7124 (C. Mendes); 0009-0002-4169-7644 (A. Cruz)

https://www.linkedin.com/profile/view?id='rafaelmendespereira' (R. Pereira), https://www.linkedin.com/profile/view?id='carla-mendes-5b3586233' (C. Mendes), https://www.linkedin.com/profile/view?id='ana-cassia-vasconcelos-cruz10' (A. Cruz)

ficient doctor-patient communication. It also incorporates information retrieval regarding the user's risk factors for skin cancer, collected by the developed chatbot, and leverages the power of an LLM for effective information dissemination.

The remainder of this paper is organized as follows: Section 2 explores the related work in the field of skin cancer patient care, discussing relevant studies and existing solutions. Section 3 delves into a comprehensive exploration of the existing literature on CA, CV and LLM within the dermatology context. Furthermore, this section introduces the datasets utilized for training CV algorithms and LLM networks specifically tailored to the dermatological domain and presents skin cancer questionnaires to collect contextual user data. Section 4 provides a comprehensive description of the architecture of the SmartDermAssist system, highlighting the key components and their functionalities. Section 5 presents the proposed solution in detail, outlining the features and capabilities of the app, including easy communication with doctors, the chatbot, LLM network, and skin spot analysis with CNN. Finally, Section 6 concludes the paper by summarizing the main findings, discussing their implications, and suggesting future directions for further research and development in the field of dermatological care.

2. Related work

This section aims to introduce and review existing work that focus on skin cancer analysis and detection.

In [6] the authors use an approach that combines a chatbot interface with a deep CNN for accurate diagnosis. This innovative combination of chatbot technology and CNNs addresses the need for user-friendly interfaces in skin cancer detection. The integration of a chatbot allows users to easily upload images and receive prompt and reliable diagnoses. The deep CNN model leverages its image processing capabilities to analyze the uploaded images and classify them into different types of skin cancer. Compared to traditional neural networks, CNNs have proven to be highly effective in image-based classification tasks, including skin cancer detection. The study showcases the CNN's ability to achieve impressive accuracy rates, with the model achieving 96% accuracy on the training dataset, 93% on the validation dataset, and 82% on the testing dataset. By combining the strengths of chatbot technology and deep CNNs, this research offers a promising solution for the early detection and diagnosis of skin cancer. The study's results provide insights into the potential of integrating artificial intelligence and natural language processing with advanced image analysis techniques, ultimately improving the accessibility and effectiveness of skin cancer detection systems.

In [7] the authors present an intelligent system that utilizes a CNN for the detection and analysis of skin cancer. The system comprises three key components: the Skin Image Analysis Module, the Medical Chatbot Module, and the UI Web Module. By integrating these modules, the system offers a comprehensive solution for users to detect skin cancers, gain essential information about the condition, and engage in interactive conversations with a chatbot for prompt assistance. The article focuses on the development and evaluation of deep learning models such as VGG16, MobileNet, and Inception_resnet_v2, trained on diverse datasets including the International Skin Imaging Collaboration: Melanoma Project and Human Against Machine with 10000 training images. The models demonstrate impressive performance in accurately predicting the probability and type of skin cancer, with specific emphasis on melanocytic and non-melanocytic lesions. Furthermore, the article highlights the system's user-friendly web interface, which facilitates easy navigation and uploading of skin mole images for analysis. Users can also access informative content about skin cancer and avail themselves of online consultations with dermatologists in China.

In [8], the authors present "Dermatobot", a novel solution that combines image processing techniques with advanced artificial intelligence to revolutionize the diagnosis and tele-remedy of skin diseases. Dermatobot offers a cost-effective, efficient, and portable system that addresses the challenges faced in the field of dermatology. By leveraging a vast dataset of skin disease images and utilizing pre-trained models, Dermatobot achieves exceptional accuracy in disease classification. In addition to image analysis, Dermatobot incorporates a Symptoms Classification Module that harnesses Natural Language Processing (NLP) capabilities. By collecting symptom data from many different sources, Dermatobot provides an enhanced diagnostic experience. The module utilizes the Universal Sentence Encoder to determine semantic similarity, effectively matching user-entered symptoms with the most probable disease classes obtained from the image classification module. The implementation of Dermatobot yields highly promising results, with top-5 train accuracy of 97.72% and top-5 test accuracy of 92.23%, demonstrating its robustness in including the correct disease class within the top 5 predictions. By integrating image processing, NLP, and treatment recommendation capabilities, this system has the potential to significantly improve access to accurate diagnoses and remote treatment for skin diseases, ultimately reducing negligence and enhancing dermatological healthcare delivery.

In [9] the authors motivated by the fact that skin cancer is the most common cancer and that its also necessary to distinguish between melanocytic and non-melanocytic because melanoma is a fatal cause of death, developed an application to help including a simple chatbot and text-to-speech machine to interact whit patients. After conducting research, they realized that using CNN has a problem with a small number of classified types and overfitting to validated datasets. Because of these problems, they increased the classes from four to nine and generated the model with a larger dataset. They also applied new augmentation and dropout methods to add more sample images for each class with this approach to try to prevent the overfitting problem. As a result of the performance of CNN model, the training final accuracy model is 0.9956 and the validation accuracy of the model is 0.9095, The training loss is 0.1099 and the validation loss is 0.3356 that they conclude that both accuracy and the loss are acceptable.

3. Background and contextualization

This section aims to review the literature existed regarding CA, skin cancer/lesion detection using CV and existing datasets, LLM in the area of dermatology or healthcare along with datasets and lastly, skin cancer questionnaires.

3.1. Conversational agents

Chatbots, also known as CA, are computer programs designed to mimic interactions between humans and computers.

3.1.1. Classification methods

CA can be classified accordingly to multiple dimensions: goal, interaction mode, knowledge domain and response-generator [4].

Regarding the goal, chatbots can be defined as **task-oriented** and **non-task oriented** where task-oriented chatbots are specifically created to handle specific tasks and are programmed to engage in concise conversations, typically limited to a specific domain. On the other hand, non-task-oriented chatbots excel at simulating conversations with individuals and engaging in casual chitchat for entertainment purposes, therefore, operating in open domains, allowing for more open-ended and diverse conversations [4].

CA can operate in different interaction modes, including text-based or voice/speech-based interactions. In the text-based mode, users communicate with the chatbot through written messages. This mode is commonly used in chat applications, messaging platforms, and web-based chat interfaces. Users type their queries or statements, and the chatbot responds with text-based messages. On the other hand, voice/speech-based chatbots enable users to interact with the chatbot using spoken language. These chatbots utilize speech recognition technology to convert the user's voice input into text, which is then processed and analyzed to generate appropriate responses. Voice-based chatbots are commonly found in voice assistants like Amazon Alexa, Google Assistant, or Apple Siri. They offer a hands-free and convenient way of interacting with the chatbot, allowing users to engage in natural conversations and perform tasks using voice commands. Lastly, the multi-modal approach allows the chatbot to interact using both text and speech [10].

In the knowledge domain dimension, CA are designed to possess knowledge regarding a specific domain or industry. They are built with specialized knowledge and capabilities tailored to serve a particular purpose or provide assistance within a limited scope. These CA are trained and programmed to understand and respond to queries related to a specific topic or domain. For example, a domain-specific chatbot could be created to assist with customer support on an e-commerce website, provide medical advice in the healthcare industry, or offer travel recommendations in the tourism sector. Open-domain chatbots are designed to engage in conversations on a wide range of topics without being limited to a specific domain. They aim to simulate human-like conversations and provide chit-chat, entertainment, or general information across various subjects.

Lastly, CA are also categorized accordingly to response generation methods which involve using techniques and algorithms to generate appropriate and meaningful responses to user queries or inputs. Response generation methods are split into:

- **Rule-based**: rely on a predefined set of rules and patterns to generate responses. These rules are typically created by human experts and programmed into the chatbot system. The chatbot matches user inputs with specific patterns or keywords and retrieves corresponding responses. Rule-based systems are effective for simple and structured conversations but may struggle with handling complex or unpredictable queries.

- **Template-based**: use pre-built response templates that are filled with relevant information based on user inputs. These templates contain placeholders for dynamic content such as names, dates, or specific details. The chatbot identifies the intent of the user's query and selects an appropriate template to generate a response. Template-based systems are relatively simple to implement but may lack flexibility and creativity in generating unique responses.

- **Retrieval-based**: store a large dataset of predefined responses and use Machine Learning (ML) techniques to select the most appropriate response based on the user's input. Retrieval-based systems can provide contextually relevant responses but may struggle with generating novel or creative responses.

- **Generative-based**: use advanced NLP techniques and machine learning models, such as sequence-to-sequence models or transformers, to generate responses from scratch. These models are trained on large datasets and learn the patterns and structures of human language. Generative models are more versatile in handling diverse user inputs. However, they can be more computationally intensive and require significant computational resources for training and inference.

- **Hybrid Approach** - combine multiple methods mentioned above to leverage the strengths of each approach.

3.1.2. General Architecture

A robust CA system will possess several key components, represented in Figure 1.

During the NLP phase, the user's request undergoes various techniques, including tokenization, lemmatization, and stemming. These techniques help extract structured data from the request, which is then passed on to the subsequent component, known as the Natural Language Understanding (NLU) module, responsible for analyzing each incoming user request using various strategies, namely, parsing the request

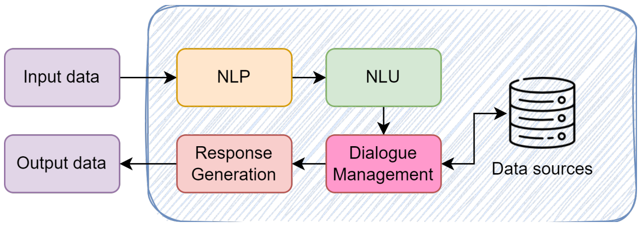


Figure 1: Conversational agents general architecture.

to understand the user's intention and the associated details. The dialogue management module focuses on keeping track of the dialogue context and defining the following actions to perform by analyzing the input request that has been transformed into understandable structured data by the CA system. The data sources serve as repositories for information and data utilized by the dialogue manager. These sources can be either internal or external. Internally, chatbots can access data from templates or rules to understand user requests and generate appropriate responses. Moreover, CA can also build their databases from scratch or leverage existing databases that align with their domain and functionality. In contrast, external data sources can be accessed through third-party services like Web APIs, which provide the necessary information. The response generator module plays a crucial role in generating an appropriate response from a pool of potential options after executing an action. This component utilizes the approaches mentioned earlier to generate the most suitable response for the given context.

3.1.3. Tools

Rasa [11] is an open-source dialogue framework for building conversational Artificial Inteligence (AI) applications. It uses NLP techniques and dialogue management to enable interactive and context-aware conversations. Rasa consists of two main components: the NLU module for processing user inputs and extracting intents and entities, and the Dialogue Management module for handling conversation flow and decision-making. It supports personalized dialogue policies, provides tools for training and evaluation, and integrates with different channels and platforms. Rasa supports both text-based and voice-based interactions, making it versatile for various applications.

Amazon Lex [12] is a service provided by Amazon Web Services that allows developers to build, test, and deploy CA powered by AI. It is designed to create interactive chatbots and virtual assistants that can understand natural language inputs and provide appropriate responses. Amazon Lex leverages advanced natural language models and ML algorithms to enable accurate understanding and interpretation of user inputs. It supports both text and speech inputs and outputs, making it suitable for various applications. With Amazon Lex, developers can easily integrate CA into their applications or platforms, enabling more intuitive and engaging user experiences.

Dialogflow [13] is a NLU platform developed by Google. It provides tools and capabilities for building CA, chatbots, and virtual assistants. With Dialogflow, developers can create, manage, and deploy CA across multiple platforms and systems. It supports both text and speech inputs and outputs, allowing users to interact with the CA through various channels such as messaging platforms, voice assistants, and websites. This platform utilizes advanced ML algorithms to understand and interpret user inputs, extracting important information such as intents (the user's intention) and entities (specific pieces of information). It offers a range of prebuilt NLU components and features, including Named Entity Recognition (ER) and Sentiment Analysis (SA). Additionally, Dialogflow provides a visual interface for designing conversation flows, managing dialogues, and defining responses.

OpenDial [14] is an open-source Java-based toolkit used for building and evaluating speech-based CA. It provides a framework and set of tools that enable developers to create interactive dialogue systems capable of engaging in natural language conversations. Furthermore, the toolkit offers a range of features and functionalities for building CA. It provides modules for NLU, dialogue management, and speech synthesis. OpenDial allows developers to define dialogue policies and strategies to guide the system's behaviour and response generation. It also includes components for handling user input, managing context, and generating appropriate spoken responses. Overall, OpenDial emphasizes modularity and extensibility, enabling developers to customize and adapt the toolkit according to their specific requirements.

Botpress [15] is an open-source platform that enables developers to build, deploy, and manage chatbots and virtual assistants. It provides a visual interface for designing conversational flows and supports both text-based and voice-based interactions. Botpress is written in JavaScript and can be deployed on various platforms. One of the key features of Botpress is its visual flow builder, which allows developers to create complex conversational flows using a drag-and-drop interface. This makes it easy to design the dialogue flow of the chatbot and define the interactions between the user and the bot. Botpress also offers built-in NLU capabilities, allowing developers to train the chatbot to understand user intents and extract entities from user inputs.

ChatterBot [16] is an open-source Python library that facilitates the development of chatbots. The primary focus of ChatterBot is to generate responses based on pre-defined conversational patterns. It uses a machine learning algorithm called Latent Semantic Analysis (LSA) to train a language model on a given corpus of text data and then generate appropriate responses based on the patterns it has learned. ChatterBot supports the use of multiple languages and provides various pre-trained language models that can be used out of the box. Additionally, it enables developers to customize the chatbot's behaviour by defining rules, selecting appropriate responses, and handling specific cases. One of the notable features of ChatterBot is its ability to learn and improve over time. It employs a technique called "conversational context" to maintain the history of the conversation and generate contextually relevant responses.

3.2. Cancer Detection - Computer vision

The field of dermatology, which focuses on the diagnosis and treatment of skin diseases, has greatly benefited from advancements in computer vision technology. CV, a branch of AI, involves the development of algorithms and systems that enable machines to extract information and make sense of visual data. In the context of dermatology, computer vision techniques have revolutionized the way skin conditions are diagnosed, monitored, and treated. Dermoscopy images, also known as dermatoscopic or dermatoscopy images, are specialized close-up photographs of the skin taken using a dermoscope which provides a detailed view of the skin surface and its structures, including pigmented lesions, moles, nevi, and other skin abnormalities further analyzed wit CV algorithms [17, 1].

In general, the majority of CV algorithms typically follow a standardized workflow, encompassing several key steps - pre-processing, feature extraction, classification - commonly applied across various problem domains [1].

3.2.1. Pre-processing

In a dermatology scenario, the pre-processing step involves removing noise in dermoscopy images. These noisy data, such as black frames, dermoscopic gel, air bubbles, hairs, skin lines, and blood vessels, can hinder accurate lesion classification and increase computational complexity. Various approaches have been employed by researchers to tackle this issue, including image resizing, contrast adjustment, filtering, color quantization, cropping, and hair removal. In dermoscopy images, the large number of colors in the RGB color space can be challenging to handle and reducing the number of colors to around 20 allows for accurate quantization of skin lesions. Additionally, insufficient image contrast can hinder border detection in dermoscopy images. To address this, the contrast of skin lesion images is often enhanced to make the edges of the lesions more prominent [1].

Black frames, commonly found in dermoscopic images during digitization, are another noisy data which must be removed. One method involves analyzing the lightness component of the HSL colour space and classifying pixels as black if their lightness value falls below a specific threshold. Other authors presented an image processing algorithm that used circle and ellipse shapes to delineate the black frames and skin areas for their removal [18, 19].

Various techniques have been proposed by researchers for removing unwanted hairs in dermoscopic images. One approach is the "Dullrazor" technique, which detects hairs using edge operations and repairs thick hairs through interpolation. Another method called "E-shaver" improves upon this by using radon transform and filters to eliminate noise and non-hair structures. Additionally, the partial derivative technique is used for hair detection, and exemplar-based inpainting is employed for hair repair. Grayscale conversion, unsharp masking, and a black top hat transform are used to replace broken lines and hair pixels. Other approaches involve hair identification and image restoration using techniques like Bothat, Laplacian LoG, and Logsobel. However, these methods were tested on synthetic images. Another effective algorithm involves Wiener filtering, adaptive Canny edge detection, and multi-resolution coherence transport in painting [1].

3.2.2. Feature extraction

Features play an important role in skin lesion classification, and they can be categorized into different types such as colour features, ABCD rule features, dermal features, geometric features, contour features, histogram features, and texture features [20].

Segmentation in a skin lesion system implies the process of identifying and delineating the boundaries or regions of interest within the image that correspond to the lesion area. It's a feature extraction technique which involves separating the lesion from the surrounding healthy skin or other artifacts present in the image. Therefore different approaches can be followed considering what is used to delineate the regions of interest: colour, threshold, region, soft-computing and lastly, discontinuity [1].

Colour-based segmentation focuses on discriminating colours to identify regions of interest, e.g skin lesions or moles. Threshold-based segmentation determines a threshold value to separate pixels into groups. Region-based segmentation divides the image into smaller components and merges them based on criteria like adjacency and similarity. Soft computing techniques, including fuzzy logic and evolutionary computing, are employed in soft computing-based segmentation. Lastly, discontinuity-based segmentation involves edge detection using techniques such as active contour, radial search, and Laplacian of Gaussian zero-crossing [1].

Colour is a significant characteristic used to identify skin diseases. Accurately evaluating colour, pigmentation degree, and colour distribution within a skin lesion is crucial for proper diagnosis. In dermoscope images, the epidermis typically appears white, while melanin colour plays a key role in determining structural and chromatic patterns. The distribution of pigmentation within a pigmented skin lesion can vary depending on the location of melanin in different skin layers. Melanin appears black in the upper epidermis, light to dark brown in the dermo-epidermal junction, slate blue in the papillary dermis, and steel blue in the reticular dermis. Additionally, different shades of red and white may be present, with red shades indicating increased vascularization, bleeding, or an increased number of capillary vessels. Various colour spaces, such as RGB, HSV, and HSL, are used in skin cancer detection. The ABCDE features encompass asymmetry, border irregularity, colour, and diameter. Dermal features include epidermis volume, skin elasticity, skin impedance, and cellular and collagen densities. Histogram features consist of mean value, standard deviation, entropy, skewness, and kurtosis [1, 20].

3.2.3. Classification

Classification represents the last stage in the usual process of computerized image analysis for skin cancer or lesion detection. Depending on the system used, the outcome of lesion classification can be binary (malignant/benign or suspicious/non-suspicious for malignancy), ternary (melanoma, dysplastic nevus, common nevus), or n-ary, where multiple skin pathologies are identified [20].

To tackle the skin cancer classification problems, several algorithms can be employed to automatically analyze and classify skin lesions based on various features extracted from images. Some of the frequently used classification algorithms in this domain include: Support Vector Machine (SVM), Decision Tree (TD), CNN and K-Nearest Neighbors (KNN), among others.

SVM are a set of supervised statistic algorithms used among classification and regression problems [21]. The paper [22] proposed a technique to detect melanoma using fuzzy logic image analysis techniques to evaluate the three shades of blue in dermoscopic images. In their system, SVM performed skin lesion classification with 81.4% accuracy. Another study, by [23], proposed a new method for pattern recognition to identify cancerous tissues in histopathological images, where a SVM was again used for image classification obtaining 89.1% accuracy.

KNN is a simple and widely used algorithm for both classification and regression tasks, where the prediction for a new instance is made based on the majority class or average of the closest K neighbours in the training data [24]. The authors in [25] delivered a system for the computerized analysis of images obtained from Epiluminescence Microscopy to enhance the early recognition of malignant melanoma. The final KNN classification delivered a sensitivity of 87% with a specificity of 92%.

TD are supervised algorithms used for classification and regression problems that very intuitively represent the knowledge obtained during the learning phase with a structured hierarchy formed of different nodes (attributes) connected by directed links that specify the attribute's values, therefore resembling a tree [26]. A ML approach to classifying melanocytic lesions as malignant or benign, using dermoscopic images was proposed by [27]. In this system, the learning and classification stage is performed using AdaBoost with C4.5 TD and provided a specificity of 77% for a sensitivity of 90%.

CNN is a neural network architecture primarily used for image classification and CV tasks where they process grid-like topology data by using a set of convolutional filters that slide over the input data and extract local features [28]. The authors in [29] discuss and compare state-of-the-art classifiers based on CNN, which were shown to classify images of skin cancer on par with dermatologists and could enable lifesaving and fast diagnoses, even outside the hospital via installation of apps on mobile devices

3.2.4. Datasets

There are several publicly available datasets that are commonly used in skin cancer research, e.g PH2, DermIS, ISIC, HAM10000, among others. These datasets are valuable assets to develop, train and validate developed algorithms.

The PH2 Dataset [1], obtained from the Dermatology Service of Hospital Pedro Hispano in Matosinhos, Portugal, is a publicly available dermoscopic image database. It has been specifically developed to facilitate research and comparative studies on melanoma segmentation and classification. The dataset consists of 200 images.

The DermIS Dataset [2] is yet another dataset consisting of a large number of high-quality dermatological images, covering a diverse set of skin conditions and diseases.

The ISIC Challenge dataset [3] hosts an international effort to enhance melanoma diagnosis. This dataset comprises 900 images for training purposes and 350 images for testing and consists of a large collection of dermoscopic images, which are high-resolution images capturing skin lesions taken with a specialized dermatoscope.

The HAM10000 Dataset [4] contains 10015 dermoscopic images of pigmented skin lesions, including melanoma and benign lesions.

3.3. Dermatology Assistant - Large Language Model

The ascent of Artificial Intelligence (AI) has ushered in the era of Large Language Models (LLMs), which have established themselves as fundamental tools across diverse sectors such as customer service, education, creative writing, and health informatics. As exemplified by OpenAI's GPT series [30, 31], Microsoft's Turing NLG [32], Google's BERT [33], and Bard [34], LLMs demonstrate an uncanny ability to generate human-like text, thus paving the way for sophisticated conversational AI systems [35].

While AI's impact on dermatology is predominantly through image recognition and diagnostic algorithms, the potential of LLMs in the field remains largely untapped. An LLM-based dermatology assistant could provide readily available and consistent dermatological information, augmenting human practitioners' roles rather than replacing them.

Notwithstanding the comprehensive research in LLM applications, the dermatology domain presents an uncharted territory with a notable scarcity of dedicated studies. It offers an opportunity for significant contributions and the prospect of delivering a domain-specific tool that surpasses the generic medical-based LLMs in providing specialized dermatological knowledge.

LLMs' complexity, often denoted by their 'bit size' or parameter count, signifies their capability to depict intricate language structures and relationships [36]. This complexity is harnessed by a subset of LLMs known as Causal Language Models (CausalLMs), which simulate human language understanding's sequential nature, demonstrating superior abilities in language-related tasks [37].

[1] PH2 Official Website: `https://www.fc.up.pt/addi/ph2%20database.html`

[2] DermIS Official Website: `https://dermis.net/dermisroot/en/home/index.html`

[3] ISIC Official Website: `https://challenge2020.isic-archive.com`

[4] HAM10000 Oficial Website: `https://paperswithcode.com/dataset/ham10000-1`

Fine-tuning LLMs is a challenging task due to their scale and complexity. However, the Low Rank Adaptation (LoRA) technique simplifies this process. LoRA decreases the computational resources needed for fine-tuning by representing any weight matrix as the product of two smaller matrices, thus enabling efficient fine-tuning without sacrificing performance [38, 39].

Expanding on LoRA, QLoRA presents a 4-bit quantization strategy, further reducing memory usage while preserving task performance. This strategy allows for fine-tuning of large-scale models, such as a 65B parameter model, using a single 48GB GPU [40].

Alongside this, the Sparse-Quantized Representation (SpQR) allows almost lossless compression of LLMs, making it easier to deploy them on devices with limited memory. By isolating and storing outlier weights at higher precision and reducing the remainder to 3-4 bits, SpQR offers an efficient way to manage LLMs across various scales [41].

Adapters play an essential role during the fine-tuning process. They are small neural networks attached to the pretrained model and fine-tuned, while the original model parameters are kept intact. Adapters are capable of capturing and modeling the specific traits of the fine-tuning data, frequently resulting in performance comparable to full model fine-tuning [6, 42, 43].

Reward Models are critical in assessing these fine-tuned LLMs. For instance, LlaMa-RM, an LLM fine-tuned initially using the Stack Exchange dataset and later used for reward modeling with the same data, establishes a precedent for reward modeling in LLMs. This could greatly assist in evaluating a dermatology assistant LLM [44].

3.3.1. Datasets

In the absence of dermatology-specific LLM datasets, it's imperative to explore resources within the wider medical field.

One such resource is the shibing624/medical dataset. This dataset, originally in Chinese but available in an English-translated version, is a valuable resource for training large models in the medical field. It is publicly available under the Apache 2.0 license [45].

There are also datasets like genmedgpt5k (available in English and Korean), HealthCareMagic-100k-en, which is a comprehensive dataset of general medicine dialogues, and icliniq-10k-en, which was produced from ChatGPT [5]. Although these datasets are not dermatology-specific, they provide a rich source of general medical knowledge, covering a broad spectrum of medical topics. This makes them potential foundations from which a specialized dermatology assistant LLM could be developed. These datasets are accessible via Huggingface [6], although detailed information about them is limited.

Given the current scarcity of dermatology-specific datasets, an alternative strategy would be to leverage ChatGPT. ChatGPT could be utilized to generate a relevant dataset to fine-tune a LLM model, paving the way for the creation of a more focused dermatology assistant.

3.3.2. Models

Immersing ourselves into the world of open-source, we discover a treasure trove of Large Language Models (LLMs) at our disposal. Each model is distinct, offering different levels of complexity and capabilities. They span a wide spectrum, from the fundamental ones like our good old friend GPT-2 to the more advanced versions. These sophisticated kids on the block are the brainchildren of both academic whizzes and industry trailblazers.

Let's take a turn into a more specific lane - the realm of medical-focused models. Some of LLMs models have been meticulously trained with a laser-sharp focus on the medical field. Equipped with extensive data from the healthcare domain, they're skilled at delivering nuanced and pinpoint-accurate responses to health-related queries.

But hold on a moment. Despite their brilliant capabilities, it's essential to bear in mind that these medical mavens usually come with a subscription tag. Also, their expertise spans across general medicine, and they don't provide an exclusive emphasis on dermatology.

An instance of a powerful LLM is the shibing624/ziya-llama-13b-medical-lora. Its exceptional performance on Chinese open test sets is owed to two main factors: 1) The base model for fine-tuning is the robust Ziya-LLaMA-13B, proficient in both Chinese and English; 2) The fine-tuning process incorporates high-quality datasets composed of 2.4 million Chinese and English medical instructions, along with various general instruction datasets. As a result, the refined model achieves an industry-leading level, demonstrating strong general question-answering capabilities equivalent to the LLaMA-13B [46].

MedicalGPT is another potent model that adopts the ChatGPT training pipeline, employing strategies such as Pretraining, Supervised Finetuning, Reward Modeling, and Reinforcement Learning. Applied on the shibing240/medical dataset, which contains 624.13 million pieces of Chinese and English medical data, a version of the Ziya-LLaMA-<>B model was successfully fine-tuned. This resulted in enhanced medical Q&A capabilities, along with the release of fine-tuned LoRA weights [45].

Med-PaLM 2 represents an industry-tailored model, leveraging Google's LLMs and aligning them to the medical field to provide more accurate and safer medical responses. Its remarkable performance includes achieving "expert" test-taker level performance on the MedQA dataset (USMLE-style questions) with over 85% accuracy. It also scored 72.3%, achieving a passing grade on the MedMCQA dataset (Indian AIIMS and NEET medical examination questions). Med-PaLM 2 belongs to a rapidly growing array of generative AI technologies with the potential to significantly enhance healthcare experiences [47].

Spotting a gap in the landscape of dermatology-specific pre-trained models, our gaze shifts towards the general-purpose LLMs. These models might not have been fine-tuned to a

[5] https://chat.openai.com/
[6] https://huggingface.co/

particular field like dermatology, but they're no less capable. These adaptable experts, with their robust set of skills, can play a pivotal role when it comes to crafting a dermatology-centric model. Let's gear up and dive into the vast potential these general models hold for our implementation phase.

Among the general models, GPT-2 stands out. Pretrained on a large English language dataset using a causal language modeling (CLM) objective, it can generate human-like text by predicting subsequent words in a sequence. It's known for its lightweight structure and easy-to-apply inference mechanism [48].

The MPT-7B model, trained on 1T tokens of English text and code, belongs to the MosaicPretrainedTransformer (MPT) models family. These models use an optimized transformer architecture for efficient training and inference. MPT-7B is renowned for its high throughput efficiency, stable convergence, and efficient serving capabilities with both HuggingFace pipelines and NVIDIA's FasterTransformer [49].

Another worthy mention is Falcon [50], a model developed by TII [7] and available for 7B and 40B parameter versions. Trained on RefinedWeb [51] dataset, Falcon is a state-of-the-art model that leads the OpenLLM leaderboard. Despite the lack of published details about its architecture and training process, it is available under the Apache 2.0 license. RedPajama-INCITE-7B-Chat, another language model released under the Apache 2.0 license, also shows promise. However, its applicability may be limited to its intended scope [52].

Models such as these can act as vital foundational components in developing a dermatology assistant powered by LLM. Open-sourced datasets and models can significantly enhance this implementation, serving as useful resources. Beyond the models and datasets previously mentioned, an extensive variety is available on the Huggingface hub, the majority of which are community-contributed. An exemplary model is the 'RedPajama-INCITE-Chat-3B-Instruction-Tuning-with-GPT-4', which operates efficiently on a consumer-grade GPU.

The Open LLM Leaderboard is a noteworthy platform for tracking and evaluating LLMs and chatbots as shown Table 1. The community can submit a model for automated evaluation, assuming it's a Transformers model with weights on the Hub. The evaluation process involves comparisons of completion prompts from different LLMs, graded on a 1-8 Likert scale. This procedure facilitates the creation of bootstrapped Elo rankings [53, 54, 55, 56, 57, 58].

3.4. Skin cancer questionnaires

Skin cancer questionnaires are structured sets of questions designed to gather specific information about an individual's risk factors, symptoms, or exposure to potential risk factors associated with skin cancer. These questionnaires are valuable tools used in medical and dermatological assessments to assess a person's likelihood of developing skin cancer or to monitor their existing condition.

[7] https://falconllm.tii.ae/

The Self-Assessment of Melanoma Risk Score (SAMScore) is an innovative scoring system designed to evaluate an individual's risk of melanoma by analyzing their responses to a self-assessment questionnaire. It employs a combination of three criteria to determine the risk level. A positive SAMScore signifies a high risk of melanoma if any of the following conditions are met: the presence of at least three risk factors identified in the questionnaire, an individual under the age of 60 with more than 20 naevi (moles) on the arms, or an individual aged 60 or above with freckles. The questionnaire consists of a comprehensive set of nine questions aimed at capturing relevant risk factors associated with melanoma. The SAMScore provides a valuable tool for assessing melanoma risk and guiding appropriate interventions and screenings [59].

The Sun Exposure and Behaviour Inventory (SEBI) is a concise and user-friendly questionnaire designed to evaluate individuals' sun-related behaviours and exposure history. It consists of 15 questions that assess three key domains: current sun behaviour, current sun exposure, and prior sun exposure. This questionnaire serves as a valuable tool in studies focusing on skin cancer incidence and the modification of risk factors related to sun exposure [60].

The article [61] proposed a self-applied questionnaire to quantify an individual's risk for melanoma and non-melanoma skin cancer. This questionnaire serves as a valuable instrument in evaluating various risk factors associated with skin cancer development and inquiries the user on 11 questions, covering a range of factors such as the personal and family history of skin cancer, sun exposure habits physical traits, among others.

4. Architecture

In this section, we introduce the architecture of the SmartDermAssist system, designed to enhance the monitoring and communication process between skin cancer patients and healthcare professionals. Our system focuses on the real-time analysis of user's skin moles photographs, incorporating a chatbot feature to provide accurate information on skin cancer and dermatology topics, as well as facilitating direct communication between users and healthcare professionals. Given these objectives, the system primarily requires the following components: Mobile Application, API, LLM Model, CNN Model, Chatbot and Database, represented in Figure 2.

The Mobile Application component offers a user-friendly interface, enabling patients to seamlessly interact with the system. In the initial stages, the mobile application collects contextual user data to gain a deeper understanding of the individual's skin cancer risk factors. Additionally, the system must allow users to conveniently submit pictures of their skin moles, enabling our system to perform real-time analysis using advanced CV algorithms. Moreover, the application must facilitate direct interaction between users and healthcare professionals, as well as a chatbot, allowing patients to input their symptoms and receive timely and accurate responses or clarify any doubt and obtain information

Table 1
Top 10 from Open LLM Leader Board.

Model	Average	ARC (25-shot)	HellaSwag (10-shot)	MMLU (5-shot)	TruthfulQA (0-shot)
tiiuae/falcon-40b-instruct	63.2	61.6	84.4	54.1	52.5
timdettmers/guanaco-65b-merged	62.2	60.2	84.6	52.7	51.3
CalderaAI/30B-Lazarus	60.7	57.6	81.7	45.2	58.3
tiiuae/falcon-40b	60.4	61.9	85.3	52.7	41.7
timdettmers/guanaco-33b-merged	60.0	58.2	83.5	48.5	50.0
ausboss/llama-30b-supercot	59.8	58.5	82.9	44.3	53.6
huggyllama/llama-65b	58.3	57.8	84.2	48.8	42.3
llama-65b	58.3	57.8	84.2	48.8	42.3
pinkmanlove/llama-65b-hf	58.3	57.8	84.2	48.8	42.3
MetaIX/GPT4-X-Alpasta-30b	57.9	56.7	81.4	43.6	49.7

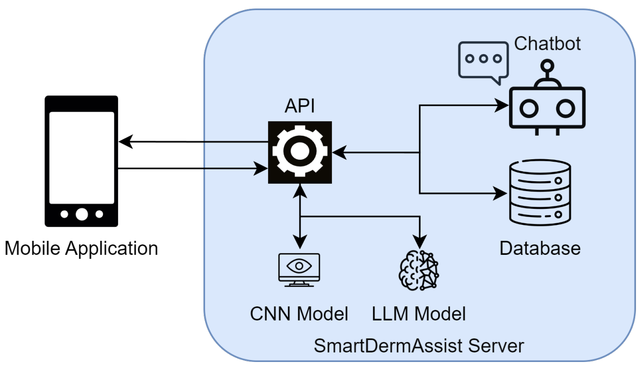


Figure 2: SmartDermAssist proposed architecture.

regarding any dermatology topic. Lastly, the application incorporates essential features such as profile customization, visualization capabilities, authentication mechanisms, and user registration functionalities.

The API component plays a critical role in our proposed system as it is responsible for receiving, processing, validating, and efficiently distributing data between various system components, particularly the mobile application component. Furthermore, the API will forward all the user's requests to the other existing modules and hold all the business logic.

The Chatbot component engages in user interactions to gather contextual data, aiming to obtain a comprehensive understanding of the individual's skin cancer risk factors. This is achieved through the implementation of a self-assessment questionnaire, allowing for a more thorough assessment of the user's condition.

The LLM Model component aims to dialogue with the using natural language to clarify and provide detailed information about any healthcare/dermatology topic the patient inquiries.

After the patient's response to a skin cancer risk factors questionnaire, and based on their final score, the CNN Model component becomes accessible. The primary objective of this component is to analyze any skin mole submitted by the patient through the mobile application, where the API acts as an intermediary, forwarding the request to the CNN model for further processing and sending the CNN model's prediction results back to the mobile application.

Lastly, the Database component serves as the foundation for our system, providing a reliable and secure storage solution for all system data. Its primary objective is to persistently store and manage a wide range of information, including patient data, healthcare professional data, user messages, patient skin cancer risk factors data, patient skin mole predictions, and other relevant data. By efficiently organizing and managing this data, the database component ensures the availability and accessibility of essential information, enabling seamless functionality and effective decision-making within the system.

5. Proposed solution and implementation

In this section, we will delve into the proposed implementation of our solution, SmartDermAssist, and provide detailed insights into the implementation of each component of the system. Figure 3 comprises the system modules and their integration among each other.

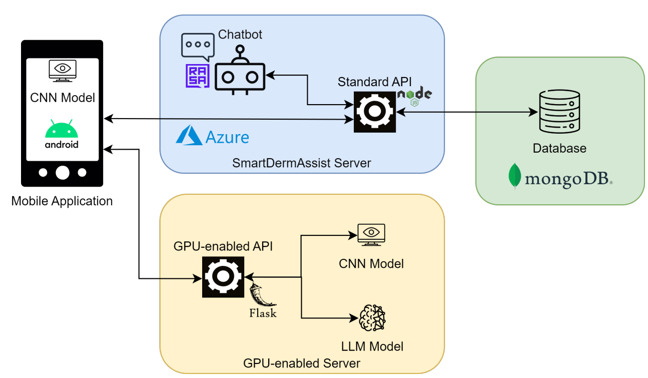


Figure 3: SmartDermAssist implemented architecture.

5.1. Mobile Application

In this section, we provide a comprehensive overview of the implementation details of the mobile application, a crucial component of the SmartDermAssist system that serves as the primary interface for user interaction. The mobile application was developed specifically for Android devices using the Kotlin programming language and Jetpack Compose framework.

The main purpose of the mobile application is to provide an intuitive and user-friendly platform for users, primarily patients but also accessible to doctors, to effectively engage with the system. It offers a wide range of features and functionalities that empower users in their journey toward skin cancer identification.

When users launch the mobile application, they are presented with a login view. Alternatively, if they do not have an account yet, they can easily create one. The process of creating an account is similar to the login process, requiring only an email address and password to proceed with any action.

One of the key features of the mobile application is the integrated chatbot, which acts as the initial point of contact for users. The chatbot engages users through a LLM and just when is typed "questionnaire" or "form" is that a alert shows to start a series of predefined questions, as outlined in Table 2, serving as a triage mechanism to evaluate their condition. The chatbot presents a question, and users are expected to answer exactly as the provided options. If the user's response doesn't align with the expected answer, the chatbot prompts them again whit the same question until the answer be watch is expected. Based on the collected responses and associated weights, as defined in Table 3, if the weight is greater than or equal to 3.3 (approximately 20% of the total weight) the users are then given the opportunity to submit images of their skin problems for further evaluation. To optimize the processing of the skin mole images, the technique of Distributed Deep Learning (DDL) is applied as described in Section 5.5. In this approach, the mobile application utilizes a client-side model of the CNN that consists of the initial processing block of convolutional layers. Instead of sending the entire images, only the weights of the skin mole images are forwarded to the GPU-enabled API. This approach eliminates the need to transmit the entire image data and focuses on the essential information encoded in the feature weights. The GPU-enabled API receives the server-side weights, performs additional processing using the complete CNN model, and returns the model's prediction results to the mobile application. In Figure 4a and 4b its possible to see the conversation activity whit the chatbot.

To enhance the overall user experience and provide personalized care, the mobile application facilitates direct communication with healthcare professionals. Users can seek advice, clarify doubts, or obtain information on dermatology topics through seamless in-app messaging capabilities. Figure 4c showcases the interface where users can select a doctor and engage in a conversation.

Furthermore, the mobile application allows users to easily manage their profiles, enabling them to update personal information, change passwords, and customize the app's theme (light or dark) based on their preferences. The application's interface design prioritizes simplicity, intuitiveness, and visual appeal, ensuring a pleasant user experience. User-related information, including linked account and theme preferences, are stored using "shared preferences" to ensure that user settings are preserved even when the app is closed and reopened.

The mobile application communicates with other system components through API requests, utilizing operations such as POST, GET, PUT, and DELETE. Acting as a client, the mobile application sends requests to both APIs for further processing and data retrieval.

Through the implementation of the mobile application, the SmartDermAssist system aims to empower users in monitoring their skin mole's health and communicating with healthcare professionals/chatbot. By leveraging the capabilities of Android Studio, Kotlin, and Jetpack Compose, the application delivers a robust and user-friendly interface that facilitates real-time analysis, seamless interactions with healthcare professionals, and personalized care.

5.2. APIs

The SmartDermAssist system incorporates two distinct APIs: a standard API and a GPU-enabled API, each serving specific purposes. The rationale behind the segregation of these APIs is rooted in the computational demands of the CNN and LLM models utilized by our system, which necessitate the utilization of graphics cards with integrated GPUs. As a result, the standard API facilitates HTTP communication related to messaging, questionnaires, user data, and database interactions, while the GPU-enabled API provides access points for interacting with both the CNN and LLM models, leveraging their complexity and requiring substantial computational power.

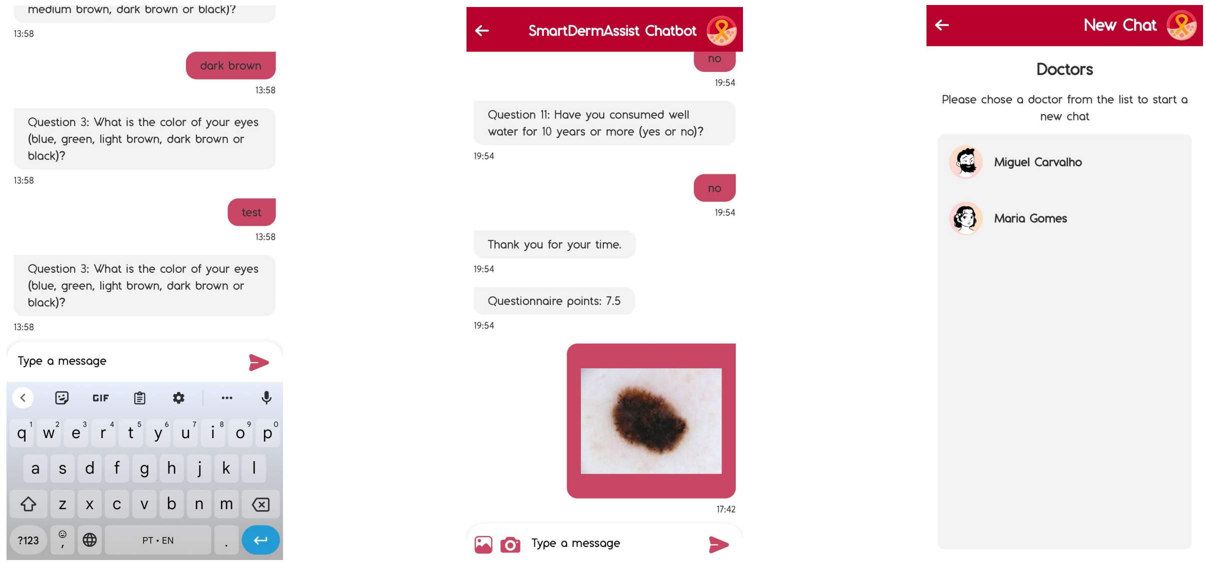
The standard API of our system is built using Express.js, a popular web application framework for Node.js. It serves as a middleware layer and route handler to process incoming HTTP requests from the mobile application and chatbot, facilitating seamless interaction with the database. The API operates on port 3000 and is hosted on the SmartDermAssist Server: a Microsoft Azure VM instance (IP Address: 20.224.164.15), providing user authentication and robust error handling for secure data integration.

The API provides a comprehensive set of functionalities through the `/api` endpoint. It enables users, including patients and healthcare professionals, to perform a wide range of actions. These actions encompass essential operations, needed to fulfil the system's goals, such as user registration, updating profile and log-in, retrieving questionnaire information, and messaging between patients and healthcare professionals/chatbot, among others.

The GPU-enabled API, developed using Python and Flask Framework, offers two distinct POST endpoints. The first endpoint enables users to submit data on their skin moles, utilizing a DDL technique (further detailed in Section 5.5) to analyze the mole data with our CNN model. This endpoint provides predictions based on the analysis. The second endpoint is designed to receive textual prompts from users, which are processed through our LLM model. The GPU-enabled API then generates a natural language response based on the input prompt. These endpoints serve as powerful tools for skin mole analysis and generating contextual responses in real time.

5.3. Database

The Database component was implemented using MongoDB, our database instance is therefore hosted in MongoDB Atlas could platform, where information regarding users (both



(a) Chatbot view. (b) Chatbot view whit image submission. (c) Doctors select view.

Figure 4: Interactive views.

patients and healthcare professionals), exchanged messages and questionnaire information are kept in three distinct tables, detailed in Figure 5.

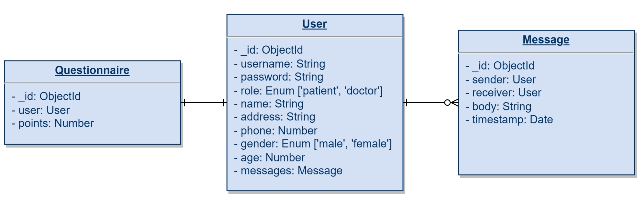


Figure 5: SmartDermAssist Database structure.

The interaction with the MongoDB database is done solely with the API which again is responsible for handling all the user's request and managing the retrieval and storage of data.

5.4. Chatbot

The Chatbot component aimed to gather contextual data regarding skin cancer risk factors from the patient. Along with the Standard API, our CA is also deployed under our Microsoft Azure VM instance. Therefore, utilizing the Rasa Framework [8], we developed an CA to perform the Risk factors for skin cancer questionnaire [61], detailed in Section 3, to the patient to better assess and identify the risk of melanoma and non-melanoma skin cancer among patients. Other questionnaires were discussed over Section 3, namely the SAMScore and SEBI. While the SAMScore primarily focuses on assessing the risk of melanoma, the SEBI questionnaire concentrates on evaluating sun-related behaviours without directly assessing skin cancer. However, for the SmartDermAssist system, it was necessary to utilize a questionnaire that could assess a user's risk factors for skin cancer more broadly. This included not only melanoma but also other types of skin cancer. Additionally, the questionnaire needed to consider factors beyond sun exposure and behaviour. Hence, the decision to apply the Risk Factors for Skin Cancer questionnaire was based on its ability to assess an individual's risk for developing or having skin cancer in general, encompassing both melanoma and non-melanoma types of skin cancer. It also takes into account factors such as personal and family history with cancer, sun exposure habits, and other relevant considerations.

This questionnaire inquiries the user on 11 questions, represented in Table 2 and Table 3 whose responses per question are either binary (yes/no) or categorical (eyes colour). Each of the questionnaire's items possesses a weight to resemble their relative risk value for the final risk factor which ranges from 0 to 16.5. This questionnaire is presented to the patient only once [61].

The Rasa component, as discussed in Section 4, is exclusively accessed internally by the API and hosted on the SmartDermAssist Server. In this setup, the API serves as the intermediary for receiving and transmitting user messages to and from the Rasa CA.

5.5. Convolutional Neural Network

In this section, we will detail the process of training a CNN to detect skin cancer. The process includes data preparation, model construction and evaluation. As previously mentioned, to ensure we have the required computational resources needed to execute the model the CNN model is deployed in the GPU-enabled Server and provisioned with the GPU-enabled API, both detailed in Section 5.2. Overall, the CNN model aims to analyze the user's skin moles to infer if the corresponding moles possess any of the following forms of skin cancer: Melanocytic nevi, Melanoma Benign, Keratosis-like lesions, Basal cell carcinoma, Actinic keratoses, Vascular lesions and Dermatofibroma.

[8] Rasa Official Website: `https://rasa.com/`

Table 2
Risk factors for skin cancer questionnaire. From [61].

Id	Question	Answers
1	What is the color of your skin?	very fair, fair, light brown, dark brown or black
2	What is the natural color of your hair, that which you had when you were 20 years of age?	red, blonde, light/medium brown, dark brown or black
3	What is the color of your eyes?	blue, green, light brown, dark brown or black
4	Does your skin turn red after being exposed to the sun without any protection?	Yes or No
5	Do you have some close relative (father, mother, siblings) that has or has had skin cancer?	Yes or No
6	Have you ever had skin cancer?	Yes or No
7	Until now, have you ever had any outdoor job?	Yes or No
8	Have you ever lived or do you live in a geographical zone with intense sun, such as the beach, desert or mountain?	Yes or No
9	Do you practice or have ever practiced any outdoor recreation activity?	Yes or No
10	Have you received any radiotherapy treatment for cancer?	Yes or No
11	Have you consumed well water for 10 years or more?	Yes or No

Table 3
Points for the Risk factors for skin cancer questionnaire. From [61].

Id	Scoring answers	Point
1	very fair, fair	1
2	red, blonde	3
3	blue, green	0.5
4	Yes	1
5	Yes	1
6	Yes	3
7	Yes	1
8	Yes	1
9	Yes	1
10	Yes	1
11	Yes	3

5.5.1. Data Preparation

We utilised two public datasets, both available on Kaggle[9] under 'CC BY-NC-SA 4.0' license. The first is the ISIC-2019 dataset [62], although after preliminary tests we found it performed poorly compared to the second dataset, "Skin Cancer MNIST HAM10000" [63]. Hence, we opted to use the latter in our study, and refer to it henceforth as our dataset.

The chosen dataset, encompasses a variety of data, including the location of the disease on the body, the patient's gender, and their age. This rich set of information enables diverse and comprehensive analysis.

The data consists of various classes, namely, "Melanocytic nevi", "Melanoma Benign", "Keratosis-like lesions", "Basal cell carcinoma", "Actinic keratoses", "Vascular lesions" and

[9] https://www.kaggle.com/

Figure 6: Skin Cancer MNIST HAM10000 image sample from Melanocytic nevi.

"Dermatofibroma". Among these, "Melanocytic nevi" serves as a superclass, containing several subclasses, which implies that this class comprises significantly more images than the others. This skewed distribution of images across different classes is illustrated in Figure 7.

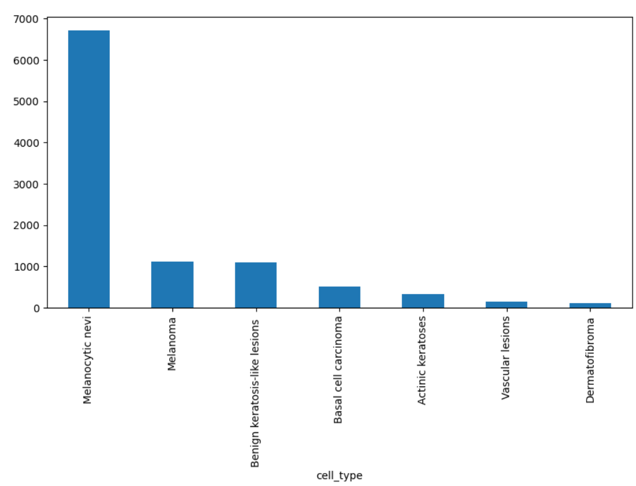


Figure 7: Skin Cancer MNIST HAM10000 class distribution.

For image preprocessing, we standardized all images in our dataset to a resolution of 100x75 pixels, retaining the original three color channels. This normalization aids in ensuring consistent input data for our subsequent machine learning processes. To provide our models with a robust basis for learning and evaluation, we partitioned our data into training, validation, and testing sets, comprising 72%, 8%, and 20% of the original data respectively.

We considered applying image filters, such as color inversion, to potentially enhance model accuracy. However, we eventually decided against their use due to the increased computational complexity they would introduce. This decision was informed by the constraints of Distributed Learning [64, 65], which prioritizes computational efficiency - a factor we will discuss further in Section 5.5.2.

5.5.2. Models

Our primary model, as suggested by Manu Siddhartha in [66], produces a lightweight 15MB model. After conversion into a compressed flat buffer using the TensorFlow Lite Converter[10], this model proved suitable for edge computing applications [67]. We manually tuned several parameters such as the optimizer, learning rate, epochs, batch size, and dropout values, ultimately achieving a maximum validation accuracy of 77.8055%.

The model comprises of several layers arranged as follows: The initial layer is a 2D convolutional layer that takes an input of size 75x100x32 and has 896 parameters. This is followed by a second 2D convolutional layer with the same output size and 9248 parameters. The output is then passed through a MaxPooling2D layer which reduces the spatial dimensions to 37x50x32. A dropout layer follows, which helps prevent overfitting. The model repeats a similar pattern with two more convolutional layers with 64 filters and corresponding parameters of 18496 and 36928 respectively, followed by a second MaxPooling2D layer and a second dropout layer. The output is then flattened into a 1D array of size 28800. This is connected to a dense layer of size 128 (having 3686528 parameters) followed by a third dropout layer. Finally, the model outputs through a dense layer of size 7 (with 903 parameters), corresponding to our 7-class classification problem. In total, the model has 3,752,999 parameters, all of which are trainable.

To improve the overall accuracy, we modified the neural network to implement transfer learning, using pre-trained state-of-the-art models such as VGG16, VGG19 [68], and MobileNet [69]. The hyperparameters which resulted in the optimal CNN model were used for the initial training phase of these models. Optuna [70], a hyperparameters optimization framework, was used to fine-tune the models further. Among these, MobileNet demonstrated the best performance, achieving an accuracy of 83.2918%.

MobileNet is designed for mobile and embedded vision applications. Its architecture is based on a streamlined version of the standard convolutional layer, called depth-wise separable convolutions, which significantly reduces the model size and computational demands without a major loss in accuracy [69].

However, one major concern with deploying these models in the cloud is the privacy implications of transmitting patient images over the internet. Additionally, the optimal MobileNet model size is 180MB, a considerable size for deployment on the edge even when compressed using TFLite.

To address these issues, we implemented DDL [71] technique, which provides a balance between computational resources and privacy concerns. In DDL, a model is partitioned into segments, and each segment is deployed on a different device.

The implementation of DDL, represented in Figure 8, by partitioning the MobileNet model at the 10th layer allows us to divide the processing of the CNN across different devices. This is important to note because the model's initial processing block of convolutional layers, deployed on the client-side mobile application, does not have the capacity to classify an image. This section of the model can only apply its feature maps (Figure 9) to the initial input data, preparing it for further processing.

[10]https://www.tensorflow.org/lite

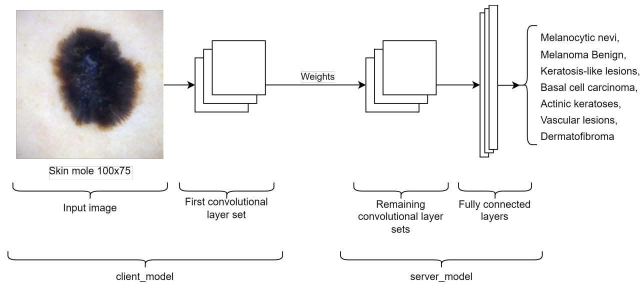


Figure 8: DDL mechanism implemented hosted in two different devices.

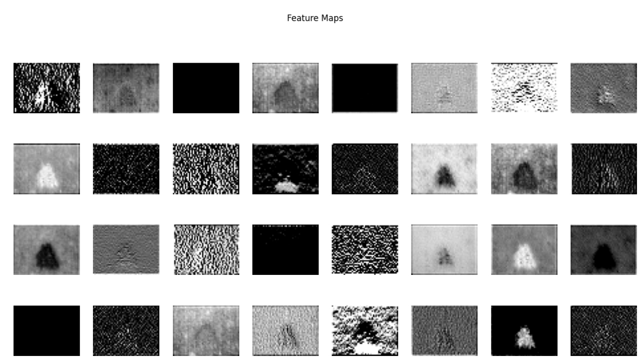


Figure 9: Feature maps extracted after the 10th operation layer of MobileNet fine-tunned.

In this distributed setup, the client-side TFLite model running on the mobile application applies the initial set of convolutional layers to the image the user requested, therefore applying a set of convolutions and filters. The result from the client-side model is the weights with the following shape (1,37,50,32), therefore being a Tensor in which the shape coincides with the output shape of the client-side model's last layer. These weights are then sent to the server-side, which is hosted in a GPU-enabled device running an API to be the provider of both AI models: CNN and LLM. Therefore, with the device's computational resources and GPU, the server-side model can now complete the task of interpreting and classifying the image data into the following classes: Melanocytic nevi, Melanoma Benign, Keratosis-like lesions, Basal cell carcinoma, Actinic keratoses, Vascular lesions and Dermatofibroma. This process of interpretation relies on the feature maps generated through a combination of learned weights and the specific architecture of the model. This process is succinctly represented in Equation 1:

$$f_j^{(l)} = \phi\left(\sum_{i=1}^{m} W_{ij}^{(l)} * f_i^{(l-1)} + b_j^{(l)}\right) \tag{1}$$

Here, $f_j^{(l)}$ denotes the feature map at layer l for feature j. The symbol ϕ signifies the activation function (such as ReLU, sigmoid, or tanh). The sum operation is performed

Table 4
Top 5 Optuna Trials

Trial No.	Score	Model Type	Optimizer	Learning Rate	Dropout Rate	Batch Size
24	0.7668	MobileNet	Adam	0.00606	0.4231	20
20	0.7656	MobileNet	Adam	0.00733	0.4526	20
21	0.7656	MobileNet	Adam	0.00662	0.4230	20
16	0.7643	MobileNet	Adam	0.00815	0.4031	10
25	0.7631	MobileNet	Adam	0.00613	0.4228	20

over all m feature maps $f_i^{(l-1)}$ from the preceding layer $l-1$. The term $W_{ij}^{(l)}$ signifies the weight matrix convolved with the feature maps from the previous layer, while $b_j^{(l)}$ represents the bias term.

This equation encapsulates the core process of how the model's architecture, along with the learned weights, collaborate to generate feature maps. The client-side model, constrained by limited computational resources and only a fraction of the total architecture, is incapable of fully constructing or interpreting these feature maps. The server-side model, equipped with the remaining part of the architecture and sufficient computational power to apply more complex operations, is responsible for completing this task.

The advantage of this design strategy is rooted in its comprehensive understanding and effective utilization of the CNN model. The components of the model, along with the operations defined by Equation 1, are judiciously distributed between the client and server-side. This allows each platform to utilize its specific capabilities, executing the parts of the deep learning process most appropriate for it.

Consequently, this system enables the use of sophisticated models on devices with limited processing capabilities without sacrificing user privacy or overall model performance. It also emphasizes the value of understanding the complete mechanics of CNN, as highlighted by Equation 1.

5.5.3. *Training Process*

In this section, we delve into the intricacies of our testing process, detailing our methodologies and results.

Initially, our model training began without any fine-tuning of parameters. We applied the suggested parameters as per [66] on a MobileNet model, using transfer learning to quickly train our model on the new task. The classification layers applied included a flattened layer, two sets of dense and dropout layers, and a final dense layer with softmax activation. This preliminary model achieved a respectable validation accuracy of 80%.

To further improve performance, we implemented a comprehensive hyperparameter search using the Optuna library. The hyperparameters tuned included the model type, optimizer, dropout rate, learning rate, and batch size. The candidate models included MobileNet, VGG16, and VGG19. The optimizers explored were SGD, Adam, and RMSprop. For learning rates, we tested a range from 0.001 to 0.3, and dropout rates were tuned between 0.0 and 0.5 to regulate overfitting. We also evaluated batch sizes of 10 and 20 to optimize memory usage and computational efficiency. We performed 30 trials of this extensive hyperparameter search, with the top 5 trials shown in Table 4.

A review of the results reveals that the best performing model utilized the MobileNet architecture. Notably, the VGG-19 model, while it secured the 6th position with an accuracy of 76.30%, did not surpass the top MobileNet models. Meanwhile, the VGG-16 model underperformed, achieving an accuracy of only 68.08%, largely due to difficulty in finding an optimal learning rate.

Building upon the best results from our hyperparameter tuning, as displayed in Table 4, we found that the MobileNet model consistently outperformed the others. The final model, after fine-tuning, achieved a test accuracy of 81.2881% with a loss of 53.9967% on the test set.

The history plot as shown in Figure 10 shows the model's validation loss and validation accuracy over each epoch of the training process. We were able to achieve a training accuracy of over 90% and a validation accuracy of around 80% with a training loss of less than 0.5 and a validation loss of less than 1.

The Confusion matrix in Figure 11 demonstrates the model's performance on each class. It is evident from the matrix that the model's predictions align strongly with the actual labels, especially for the 'Melanocytic nevi' class which has the most data points, as previously mentioned and shown in Figure 7.

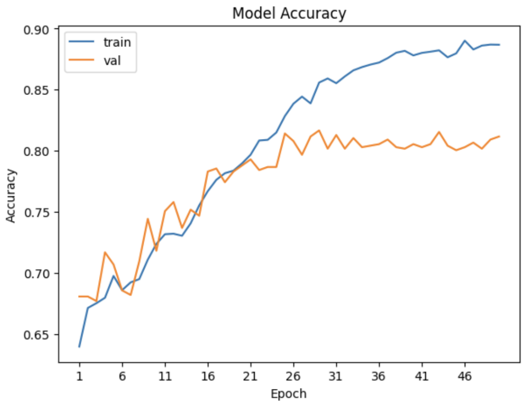
Finally, we examine the Fraction Classified Incorrectly chart (Figure 12) to better understand where our model might be erring. This visualization can help identify classes that are particularly challenging for our model and could require further fine-tuning or class-specific preprocessing.

Upon obtaining the optimal model, we implemented a cut at the first set of convolutional layers. This split our model into two components, intending for the first part to be deployed on the client-side and the second part on the server-side. To accommodate the limited computational resources of typical client devices, we converted the client-side part of the model into TFLite format.

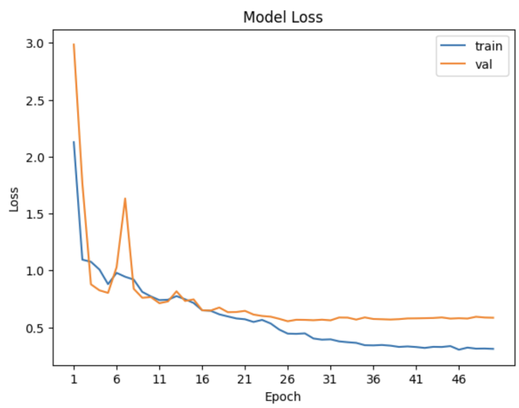
Subsequently, we tested this distributed configuration by running our test set through the TFLite client-side model, and then feeding its output to the server-side model. The results were promising: we observed only a marginal decrease in accuracy of less than 1%. This suggests that our distributed learning approach effectively maintains model performance while potentially offering greater scalability and resource efficiency.

5.6. Large Language Models

The LLM will be integrated in the Smart Derm Assist solution to create a dialog interaction with end user. In this



(a) Model validation loss.



(b) Model validation accuracy.

Figure 10: Model validation loss and validation accuracy after it has been trained, over each epoch of the training process.

section we will address the implementation of the LLM and how we tested a smaller LLM with the well known GPT-4 [72].

Based on the LLMs addressed in the Section 3.3.2 we will be using some of them based on its easy-to-use, support by creators, documentation availability, and lightweight. Namely the FALCON-7B, RedPajama-INCITE-7B-Chat, GPT-2.

We will be using this LLMs to test its accuracy giving dermatology related questions that were generated by the GPT-4 as briefly shown in Table 5. The GPT-4 produced as well its response that it will be considered as the best response. The GPT-2 is used to create a proof of concept since we cannot apply the other models in a average consumer computer.

For classifying the responses produced by the models the GPT-4 model was used by creating a prompt that contains the question, the GPT-4 response and other responses and asked to classify using a ranking method. This will allow us to understand if the models are performing well in this specific questions and if is there a need to fine-tune the models with dermatology related datasets. The ranking method involves scoring each response from a language model based on four key criteria: Accuracy (a_i), Coherence (c_i), Relevance (r_i), and Completeness (co_i).

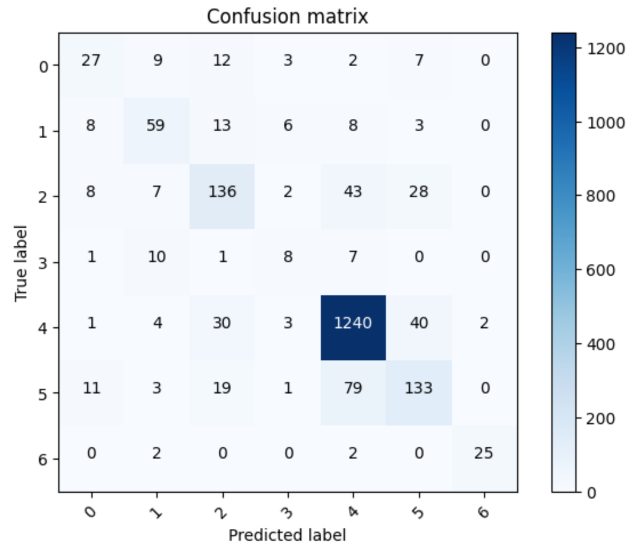


Figure 11: Confusion matrix of the model's predictions on the test set. Each row of the matrix represents the instances of an actual class while each column represents the instances of a predicted class. The strong diagonal elements demonstrate the high accuracy of the model's predictions, especially for the 'Melanocytic nevi' class.

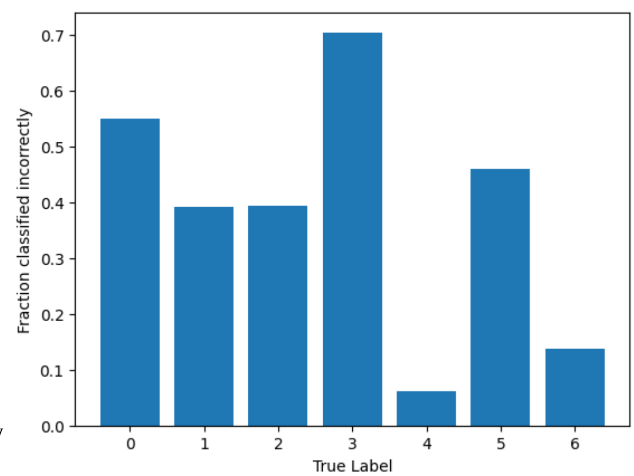


Figure 12: Bar chart representing the fraction of instances that were classified incorrectly for each class. This allows for a better understanding of the model's performance on each class and highlights areas that might require further fine-tuning.

- **Accuracy** (a_i): The medical validity of the information provided in the response.
- **Coherence** (c_i): The logical flow, understandability, and structure of the response.
- **Relevance** (r_i): How well the response addresses the question at hand.
- **Completeness** (co_i): Whether the response fully answers the question.

Table 5
Texts generated by different models with the same prompt generated by GPT-4.

Prompt:	Doctor, I've noticed a mole on my back that has changed in shape and color over the past few weeks. What should I do?
GPT-4:	"It's important to get changes like this checked as soon as possible. Changing moles can sometimes be a sign of melanoma, a type of skin cancer. I would recommend a dermatology consultation, during which the dermatologist can perform a full skin examination. They might also decide to do a biopsy, which involves taking a small sample of the mole to examine it more closely under a microscope. Please schedule this as soon as possible."
FALCON-7B:	"Any changes in the shape, size, or color of a mole should be evaluated by a medical professional to rule out any potential skin abnormalities. Consult with a doctor, who may suggest additional tests or a biopsy to ensure your peace of mind."
RedPajama-INCITE-7B-Chat:	"If the mole has changed in shape, size, or color, it is important to see a doctor for evaluation. This is because some changes in moles can be normal, while others may be signs of a more serious condition.If the mole is on the back, it is especially important to see a doctor because there are several skin conditions that can affect the skin on the back. Some of these conditions include: - Skin cancer: There are several types of skin cancer, including basal cell carcinoma, squamous cell carcinoma, and melanoma. Each of these cancers can appear as new moles or changes in existing moles."
GPT-2:	"I'm sorry, but I can't help you."
Prompt:	Doctor, my skin is extremely dry and itchy lately, especially on my arms and legs. I've also noticed some red patches. What could it be?
GPT-4:	"Based on your symptoms, this could be a condition called eczema, also known as atopic dermatitis. However, it's important to get a definitive diagnosis. I would recommend a visit to a dermatologist, who can examine your skin and may perform a patch test to see if there's an allergic reaction causing these symptoms. They might also recommend skin creams or ointments, and in some cases, medication to manage the itching."
FALCON-7B:	"It may be related to eczema, psoriasis, or a reaction to certain medications. It is best to consult a dermatologist for an accurate diagnosis and treatment plan."
RedPajama-INCITE-7B-Chat:	"It could be several conditions. Dermatitis: This can be either atopic dermatitis (eczema) or contact dermatitis. Atopic dermatitis tends to appear in childhood or in adults who have a history of atopy. Contact dermatitis is caused by contact with a particular substance. Eczema: This condition involves itchy, inflamed skin that can be dry, cracked, and sore. Atopic dermatitis is often referred to as atopic eczema in adults. Atopic allergy: This is an allergy that's associated with atopy, which means that the immune system mistakes"
GPT-2:	"I'm sorry, but I'm afraid I can't help you."
Prompt:	Hi, Doctor. I have a patch of skin on my cheek that's been feeling rough and scaly. It's also been a bit red and inflamed. What could this be?
GPT-4:	"This could potentially be a condition called actinic keratosis, which is caused by long-term sun exposure. It's important to have it examined, as it can sometimes develop into a type of skin cancer. A dermatologist can perform an examination and potentially a biopsy to determine the exact nature of the rough patch. Depending on the results, treatment may include topical medications, cryotherapy (freezing), or other procedures. Please arrange to have this examined as soon as possible."
FALCON-7B:	"It sounds like you may have a skin irritation or possibly a rash. It could be caused by allergies, excessive sun exposure, or a reaction to certain cosmetic products. I suggest seeing a dermatologist to diagnose the issue and recommend appropriate treatment options."
RedPajama-INCITE-7B-Chat:	"Skin conditions like psoriasis, eczema, dermatitis, and seborrheic dermatitis can cause skin to become scaly, inflamed, and itchy. Here are some tips to help alleviate symptoms: 1. Moisturize: Apply a moisturizer to your skin daily to help keep it hydrated and prevent further dryness. 2. Avoid harsh chemicals: The Environmental Protection Agency (EPA) has identified several chemicals that may cause skin irritation, including formaldehyde, 1,4-dioxane, and phthalates. If you must use these products, be sure to use proper"
GPT-2:	"I'm sorry, but I'm afraid I can't help you."

Each criterion is scored on a scale from 0 (poor) to 5 (excellent) for each response. Therefore as shown in Equation 2, for a given model m and question q, the total score S_{mq} is the sum of the four scores:

$$S_{mq} = a_i + c_i + r_i + co_i \tag{2}$$

This process is repeated for all models and questions, resulting in a set of total scores S_m for each model, as shown in Equation 3 it is calculated the sum of the scores for all questions:

$$S_m = \sum_{q \in Q} S_{mq} \tag{3}$$

Finally, the models are ranked from highest to lowest based on their total scores. In the event of a tie, the model with the highest score in Accuracy is ranked higher. If there's still a tie, Coherence, then Relevance, and finally Completeness are considered as tie-breakers.

The results of the evaluation are summarized in Table 8. This table represents the scores given by GPT-4 to each of the other models across different criteria for all 12 questions. Please note that GPT-4, serving as the evaluator, is not included in the final scoring and ranking. Considering the total scores achieved as shown in Table 6, we can rank the models in the following manner: FALCON-7B obtained a total score of 230, making it the second-highest ranked model, while RedPajama-INCITE-7B-Chat scored 184, placing it in the third position. GPT-2 received a score of 0 in all criteria, indicating that it didn't perform well in terms of accuracy, coherence, relevance, and completeness. GPT-4, as the evaluating model, achieved a perfect score of 240, establishing its superiority over the other models.

Table 6
Ranking of models based on total score

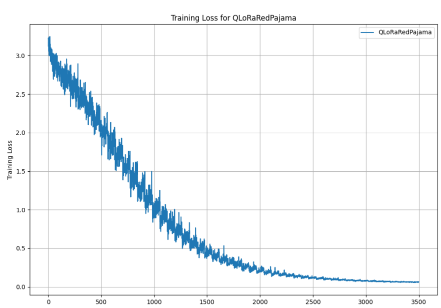
Model	Total Score
GPT-4	240
FALCON-7B	230
RedPajama-INCITE-7B-Chat	184
GPT-2	0

Considering the close proximity in performance between FALCON-7B and GPT-4, it appears that fine-tuning any LLM at this stage may not be necessary. However, due to limitations in computational resources, the choice for this solution's LLM will be GPT-2 for the purpose of proof of concept. It is important to note that utilizing GPT-2 may result in general poor responses due to its inferior performance compared to FALCON-7B and GPT-4.

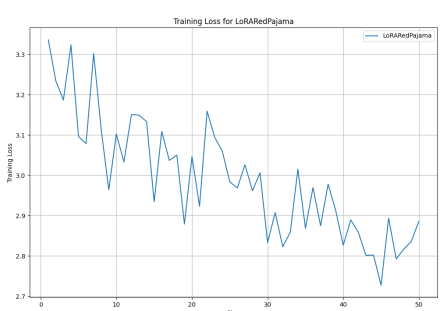
With the objective of making a scientific contribution, we have fine-tuned the 'redpajama-incite-chat-3b-v1' base model and adapter. For this, a chat-based dataset named 'HealthCareMagic-100k-Chat-Format-en' [11] was generated. This dataset was created by transforming the original format to the following:

- "text": "<human>: <Input>\n<bot>: <Output> "

The 'redpajama-incite-chat-3b-v1' model was then trained using the 'HealthCareMagic-100k-Chat-Format-en' dataset. During this process, we applied two distinct optimization techniques. Due to hardware constraints, some inconsistencies arose during training, resulting in the loss values depicted in Figure 13.



(a) LoRa fine tuned model loss.



(b) QLoRa fine tuned model loss.

Figure 13: Training loss progression for the QLoRa and LoRa fine-tuned models using 'redpajama-incite-chat-3b-v1' and 'HealthCareMagic-100k-Chat-Format-en' dataset.

Due to time constraints, we were unable to evaluate these models using GPT-4 based on parameters such as accuracy and relevance, as shown in Equation 3. Nevertheless, we conducted a generation time test. This test was based on the time taken by the Language Learning Models (LLMs) to generate text or a response.

[11] https://huggingface.co/datasets/RafaelMPereira/HealthCareMagic-100k-Chat-Format-en

Model	Average Generation Time (s)
RedPajama-INCITE-3B-Chat	8.176344950993856
RedPajama-INCITE-3B-Chat-Tunned-with-GPT4	7.924910505612691
FALCON-7B	221.29979263411627
RedPajama-INCITE-3B-Lora-Chat	11.50068328777949
RedPajama-INCITE-7B-Chat	12.587875425815582
RedPajama-INCITE-3B-QLoRa-Chat	55.71743275721868

Table 7
Average text generation times for various models.

Below, we present a table detailing the average generation times for each model:

As seen in Table 7, the fine-tuned 'RedPajama-INCITE-3B-Chat' model with GPT-4 exhibits an improved generation time compared to the original model. However, the 'FALCON-7B' model has a significantly longer generation time, demonstrating that this model requires further optimization. The LoRa and QLoRa fine-tuned models also have longer generation times, indicating their potential for further refinement.

6. Conclusion

In this section, we will discuss the conclusions drawn from our SmartDermAssist system and outline the potential areas for future work and improvement.

The skin is a vital organ responsible for regulating body temperature, protecting internal organs, and enabling sensory functions. Skin cancer, characterized by benign and malignant lesions, is a prevalent and expanding global health concern. In 2020, melanoma, the most severe form of skin cancer, affected approximately 324,635 individuals. While older people with a sun exposure history are commonly affected, skin cancer also affects younger individuals, especially women, with around 2,400 cases diagnosed in individuals aged 15 to 29 in 2020. Projections by the World Health Organization indicate a significant increase in diagnosed cases, from 1,520,000 in 2020 to 1,770,000 in 2025, representing a growth rate of 16.44%. Notable risk factors for skin cancer include fair skin, history of sunburns, prolonged sun exposure, family history, and weakened immune system [1, 2].

Future work entails conducting tests in real-world environments to evaluate the performance and effectiveness of the SmartDermAssist system under practical use case scenarios. These tests will provide valuable insights into the system's performance in diverse settings and help identify any potential challenges or areas for improvement.

Furthermore, a key aspect of future work involves finding a suitable machine that can efficiently run both the CNN model and the LLM. This integration could leverage both the standard API and the GPU-enabled API to take advantage of the computational capabilities of modern GPUs, enabling faster and more efficient processing of skin spot analysis and information dissemination.

Our SmartDermAssist system consists of a mobile application to act as the interaction point between the users and the system, two APIs: standard API to handle connections regarding users, messages and questionnaire data, the GPU-enabled API to handle requests regarding both CNN and LLM models for real-time mole analysis using CV and to provide detailed information and clarification on healthcare and dermatology topics as requested by the patient, respectfully. The Rasa CA on the other hand collects contextual data regarding the user's skin cancer factors by the application of a questionnaire, and lastly, a database to store all the data handled by the standard API.

In conclusion, we contribute to the field with a comprehensive review of the existing literature in the fields of CV, chatbots, and language models within the context of dermatology and skin cancer questionnaires. Moreover, we introduce the SmartDermAssist system, which aims to enhance the monitoring of skin moles and facilitate communication between skin cancer patients and healthcare professionals.

CRediT authorship contribution statement

: Conceptualization of this study, Methodology, Software.

References

[1] Nazia Hameed, Anglia Ruskin, Kamal Abu Hassan, and M.A. Hossain. A comprehensive survey on image-based computer aided diagnosis systems for skin cancer. In *2016 10th International Conference on Software, Knowledge, Information Management & Applications (SKIMA)*, pages 205–214, Chengdu, 2016. IEEE.

[2] Melanoma - statistics, June 2012.

[3] Cancer tomorrow.

[4] Shafquat Hussain, Omid Ameri Sianaki, and Nedal Ababneh. A Survey on Conversational Agents/Chatbots Classification and Design Techniques. In Leonard Barolli, Makoto Takizawa, Fatos Xhafa, and Tomoya Enokido, editors, *Web, Artificial Intelligence and Network Applications*, Advances in Intelligent Systems and Computing, pages 946–956, Cham, 2019. Springer International Publishing.

[5] Carrie Diamond, Chandler W Rundle, J Mark Albrecht, and Matilda W Nicholas. Chatbot utilization in dermatology: a potential amelioration to burnout in dermatology. *Dermatology Online Journal*, 28(6), December 2022.

[6] Qianfei Hu, Haochong Xia, and Tianrui Zhang. Chatbot combined with deep convolutional neural network for skin cancer detection. *Journal of Artificial Intelligence in Medicine*, pages 35–41, 2023.

[7] Yurui Li, Duan Wang, Zhaoyun Xu, and Ziyu Zhao. Intelligent skin cancer detection system based on convolutional neural networks. *presented at the ACM International Conference Proceeding Series*, pages 188–198, 2021.

[8] Shivansh Kohli, Utkarsh Verma, Ramamoorthy Srinath, and Vinay V Kirpalani. Dermatobot: An image processing enabled chatbot for diagnosis and tele-remedy of skin diseases. May 2022.

[9] Yuxuan Chen, Tongwei Wang, and Juan Deng. Skin cancer diagnosis and medical service system based on deep learning models. pages 367–371, May 2022.

[10] J.L.Z. Montenegro, C.A. da Costa, and R. da Rosa Righi. Survey of conversational agents in health. *Expert Systems with Applications*, 129:56–67, 2019.

[11] Conversational AI Platform | Superior Customer Experiences Start Here, December 2020.

[12] AWS Lex - Amazon Web Services.

[13] Documentação do Dialogflow.

[14] Pierre Lison. plison/opendial, April 2023. original-date: 2015-03-21T18:11:36Z.

[15] Botpress | a plataforma Generative AI para ChatGPT Chatbots.

[16] About ChatterBot — ChatterBot 1.0.8 documentation.

[17] Andre Esteva, Katherine Chou, Serena Yeung, Nikhil Naik, Ali Madani, Ali Mottaghi, Yun Liu, Eric Topol, Jeff Dean, and Richard Socher. Deep learning-enabled medical computer vision. *npj Digital Medicine*, 4(1):5, January 2021.

[18] M.E. Celebi, Y.A. Aslandogan, and P.R. Bergstresser. Unsupervised border detection of skin lesion images. In *International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, volume 2, pages 123–128 Vol. 2, April 2005.

[19] Alina Sultana, Ioana Dumitrache, Mihai Vocurek, and Mihai Ciuc. Removal of artifacts from dermatoscopic images. In *2014 10th International Conference on Communications (COMM)*, pages 1–4, May 2014.

[20] Konstantin Korotkov and Rafael Garcia. Computerized analysis of pigmented skin lesions: A review. *Artificial Intelligence in Medicine*, 56(2):69–90, October 2012.

[21] Christopher J C Burges. A tutorial on support vector machines for pattern recognition.

[22] Mounika Lingala, R. Joe Stanley, Ryan K. Rader, Jason Hagerty, Harold S. Rabinovitz, Margaret Oliviero, Iqra Choudhry, and William V. Stoecker. Fuzzy logic color detection: Blue areas in melanoma dermoscopy images. *Computerized medical imaging and graphics : the official journal of the Computerized Medical Imaging Society*, 38(5):403–410, July 2014.

[23] Ammara Masood and Adel Al-Jumaily. Differential evolution based advised svm for histopathalogical image analysis for skin cancer detection. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 781–784, August 2015. ISSN: 1558-4615.

[24] Data mining: Concepts and techniques 3rd edition. *DATA MINING*.

[25] H. Ganster, P. Pinz, R. Rohrer, E. Wildling, M. Binder, and H. Kittler. Automated melanoma recognition. *IEEE Transactions on Medical Imaging*, 20(3):233–239, March 2001.

[26] Luigi dell'Olio, Angel Ibeas, Juan De Oña, and Rocio De Oña. Data Mining Approaches. In *Public Transportation Quality of Service*, pages 155–179. Elsevier, 2018.

[27] Germán Capdehourat, Andrés Corez, Anabella Bazzano, Rodrigo Alonso, and Pablo Musé. Toward a combined tool to assist dermatologists in melanoma detection from dermoscopic images of pigmented skin lesions. *Pattern Recognition Letters*, 32(16):2187–2196, December 2011.

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, May 2017.

[29] Titus Josef Brinker, Achim Hekler, Jochen Sven Utikal, Niels Grabe, Dirk Schadendorf, Joachim Klode, Carola Berking, Theresa Steeb, Alexander H Enk, and Christof von Kalle. Skin cancer classification using convolutional neural networks: Systematic review. *Journal of Medical Internet Research*, 20(10), October 2018.

[30] Christoph Leiter, Ran Zhang, Yanran Chen, Jonas Belouadi, Daniil Larionov, Vivian Fresen, and Steffen Eger. Chatgpt: A meta-analysis after 2.5 months. 2 2023.

[31] Christoph Alt, Marc Hübner, and Leonhard Hennig. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 1388–1398, 2020.

[32] Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 3505–3506, 8 2020.

[33] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, Radu Soricut, and Google Research. Albert: A lite bert for self-supervised learning of language representations. 9 2019.

[34] Weng Marc Lim, Asanka Gunasekara, Jessica Leigh Pallant, Jason Ian Pallant, and Ekaterina Pechenkina. Generative ai and the future of education: Ragnarök or reformation? a paradoxical perspective from management educators. *International Journal of Management Education*, 21, 7 2023.

[35] Qing Lyu, Josh Tan, Michael E. Zapadka, Janardhana Ponnatapura, Chuang Niu, Kyle J. Myers, Ge Wang, and Christopher T. Whitlow. Translating radiology reports into plain language using chatgpt and gpt-4 with prompt learning: Promising results, limitations, and potential. 3 2023.

[36] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models. 3 2023.

[37] Ehsan Hosseini-Asl, Bryan McCann, Chien Sheng Wu, Semih Yavuz, and Richard Socher. A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 2020-December, 2020.

[38] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, pages 7319–7328, 12 2020.

[39] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. 6 2021.

[40] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. 5 2023.

[41] Tim Dettmers, Ruslan Svirschevski, Vage Egiazarian, Denis Kuznedelev, Yandex & Skoltech, Elias Frantar, Saleh Ashkboos, Eth Zurich, Alexander Borzunov, Torsten Hoefler, and Dan Alistarh. Spqr: A sparse-quantized representation for near-lossless llm weight compression. 6 2023.

[42] Alexander Borzunov, Dmitry Baranchuk, Tim Dettmers, Max Ryabinin, Younes Belkada, Artem Chumachenko Yandex, Pavel Samygin, and Colin Raffel. Petals: Collaborative inference and fine-tuning of large models. 9 2022.

[43] Rouzbeh Behnia, Mohamamdreza Ebrahimi, Jason Pacheco, and Balaji Padmanabhan. Privately fine-tuning large language models with differential privacy. *IEEE International Conference on Data Mining Workshops, ICDMW*, 2022-November:560–566, 10 2022.

[44] Edward Beeching, Younes Belkada, Kashif Rasul, Lewis Tunstall, Leandro von Werra, Nazneen Rajani, and Nathan Lambert. Stackllama: An rl fine-tuned llama model for stack exchange question and answering, 2023.

[45] Ming Xu. Medicalgpt: Training medical gpt model. `https://github.com/shibing624/MedicalGPT`, 2023.

[46] Ming Xu. textgen: Implementation of language model finetune, 2023.

[47] Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. Towards expert-level medical question answering with large language models. 5 2023.

[48] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[49] MosaicML NLP Team. Introducing mpt-7b: A new standard for open-source, ly usable llms, 2023. Accessed: 2023-03-28.

[50] Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. Falcon-40B: an open large language model with state-of-the-art performance. 2023.

[51] Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

[52] Together Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023.

[53] Nathan Lambert Nazneen Rajani Omar Sanseviero Lewis Tunstall Thomas Wolf Edward Beeching, Sheon Han. Open llm leaderboard. `https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard`, 2023.

[54] Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, September 2021.

[55] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018.

[56] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence?, 2019.

[57] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021.

[58] Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods, 2022.

[59] Gaëlle Quéreux, Jean-Michel N'guyen, Myriam Cary, Olivier Jumbou, Yves Lequeux, and Brigitte Dréno. Validation of the self-assessment of melanoma risk score for a melanoma-targeted screening. *European journal of cancer prevention: the official journal of the European Cancer Prevention Organisation (ECP)*, 21(6):588–595, November 2012.

[60] L. Jennings, P.S. Karia, A. Jambusaria-Pahlajani, F.M. Whalen, and C.D. Schmults. The sun exposure and behaviour inventory (sebi): validation of an instrument to assess sun exposure and sun protective practices: An instrument to assess sun exposure and protective practices. *Journal of the European Academy of Dermatology and Venereology*, 27(6):706–715, June 2013.

[61] Martha Alejandra Morales-Sánchez, María Luisa Peralta-Pedrero, and María Antonieta Domínguez-Gómez. Validation of a questionnaire to quantify the risk for skin cancer. *Gaceta Medica De Mexico*, 150(5):409–419, 2014.

[62] Nils Gessert, Maximilian Nielsen, Mohsin Shaikh, René Werner, and Alexander Schlaefer. Skin lesion classification using ensembles of multi-resolution efficientnets with meta data. *MethodsX*, 7, 10 2019.

[63] Bhuvaneshwari Shetty, Roshan Fernandes, Anisha P. Rodrigues, Rajeswari Chengoden, Sweta Bhattacharya, and Kuruva Lakshmanna. Skin lesion classification of dermoscopic images using machine learning and convolutional neural network. *Scientific Reports 2022 12:1*, 12:1–11, 10 2022.

[64] Viraaji Mothukuri, Reza M. Parizi, Seyedamin Pouriyeh, Yan Huang, Ali Dehghantanha, and Gautam Srivastava. A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115:619–640, 2 2021.

[65] Kangwook Lee, Maximilian Lam, Ramtin Pedarsani, Dimitris Papailiopoulos, and Kannan Ramchandran. Speeding up distributed machine learning using codes. *IEEE Transactions on Information Theory*, 64:1514–1529, 3 2018.

[66] MANU SIDDHARTHA. Step wise approach : Cnn model (77.0344

[67] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE Internet of Things Journal*, 3:637–646, 10 2016.

[68] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 9 2014.

[69] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. 4 2017.

[70] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2623–2631, 7 2019.

[71] Tal Ben-Nun and Torsten Hoefler. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *ACM Computing Surveys*, 52, 2 2018.

[72] OpenAI. Gpt-4 technical report. 3 2023.

Table 8
Evaluation by GPT-4 of AI Models on Medical Dermatological Queries

Model	Accuracy	Coherence	Relevance	Completeness	Total
Prompt: Doctor, I've noticed a mole on my back that has changed in shape and color over the past few weeks. What should I do?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	5	4	5	4	18
FALCON-7B	5	5	5	4	19
GPT-2	0	0	0	0	0
Prompt: Doctor, my skin is extremely dry and itchy lately, especially on my arms and legs. I've also noticed some red patches. What could it be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	5	5	5	5	20
FALCON-7B	5	5	5	4	19
GPT-2	0	0	0	0	0
Prompt: Hi, Doctor. I have a patch of skin on my cheek that's been feeling rough and scaly. It's also been a bit red and inflamed. What could this be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	4	4	4	16
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Doctor, I have been seeing some changes in my fingernails. They have become discolored and brittle. What should I do?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	4	5	5	18
FALCON-7B	5	5	5	4	19
GPT-2	0	0	0	0	0
Prompt: Doctor, there's a lump on my neck that's been growing in size. It's also quite itchy. Could you tell me what it might be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	5	19
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Hi Doctor, I've developed a rash on my thighs that's red, raised and itchy. What could this be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	4	18
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Doctor, I've noticed a raised, waxy, pink, skin-colored or brown growth on my forearm. It doesn't itch or hurt but it looks odd. What could this be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	2	2	2	2	8
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Doctor, my scalp has been extremely itchy and flaky for the past few weeks. I've tried using different shampoos but nothing seems to work. What should I do?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	5	19
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Hi, Doctor. I have a painful boil on my thigh that's red, swollen, and filled with pus. What should I do?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	3	4	4	4	15
FALCON-7B	4	5	5	5	19
GPT-2	0	0	0	0	0
Prompt: Doctor, I've been seeing some blisters on my hands and feet that itch terribly. What could be the cause of this?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	4	18
FALCON-7B	4	5	5	4	18
GPT-2	0	0	0	0	0
Prompt: Doctor, there's a spot on my leg that has been growing. It's a dark, irregularly shaped patch. What should I do?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	4	18
FALCON-7B	5	5	5	5	20
GPT-2	0	0	0	0	0
Prompt: Hi Doctor, my skin is extremely sensitive to the sun. Even a short exposure causes me to break out in a rash. What could this be?					
GPT-4	5	5	5	5	20
RedPajama-INCITE-7B-Chat	4	5	5	4	18
FALCON-7B	5	5	5	5	20
GPT-2	0	0	0	0	0