CLIP vs. the Classics: Augmenting Traditional Classifiers with Modern Feature Representations

Joshua Levine, Mario Aranda
University of Illinois Urbana Champaign
{joshua45, marand7}@illinois.edu

In our project, we will first compare zero-shot CLIP classification to models considered state-of-the-art over a decade ago. Second, we will see if using CLIP features with these models can outperform ResNet50 on medical image classification tasks. Our work will study two datasets: HAM10000 for skin lesion classification and the NIH Chest X-ray dataset.

The state-of-the-art approaches to these classification tasks in the 2000s used random forests and SVMs combined with useful, feature selection techniques. We will re-implement these to conduct our tests. While these methods alone cannot stand up to modern deep learning techniques, combining older models, including SVMs, decision trees, and clustering with CLIP feature extraction, will help them compete with newer techniques.

We chose datasets that would have been relevant when the older papers we were examining were written. X-ray segmentation was one of the early applications of computer vision, and skin lesion classification became relevant later. The Chest X-ray dataset labels images with classes corresponding to 14 diseases and a "No Findings" label. The diseases include Edema, Fibrosis, Pneumonia, and Hernia. The HAM10000 dataset contains dermatoscopic images of skin lesions with similar labels for seven diseases, including Actinic keratoses, basil cell carcinoma, and melanoma. We hope each domain will provide helpful information about the efficacy of the aforementioned models on medical data.

Our results showed that using CLIP and MedCLIP features boosted performance, enabling the older models to achieving almost as high accuracy as ResNet, as seen in table 1. We believe that this technique offers a valuable alternative to deep neural networks when runtime efficiency is prioritized over accuracy.

Features	Linear Probe	SVM	K-Means	Random Forest	ResNet50
Pixel Values	N/A	N/A	N/A	N/A	86.926
CLIP	**81.737**	71.457	76.347	**71.457**	N/A
MedCLIP-ResNet50	73.054	75.948	76.347	68.463	N/A
MedCLIP-ViT	74.152	**76.946**	**76.447**	70.060	N/A

Table 1. The accuracy on **HAM10000** test set for each model using each set of features. The CLIP features are the embeddings computed from each of the CLIP models.

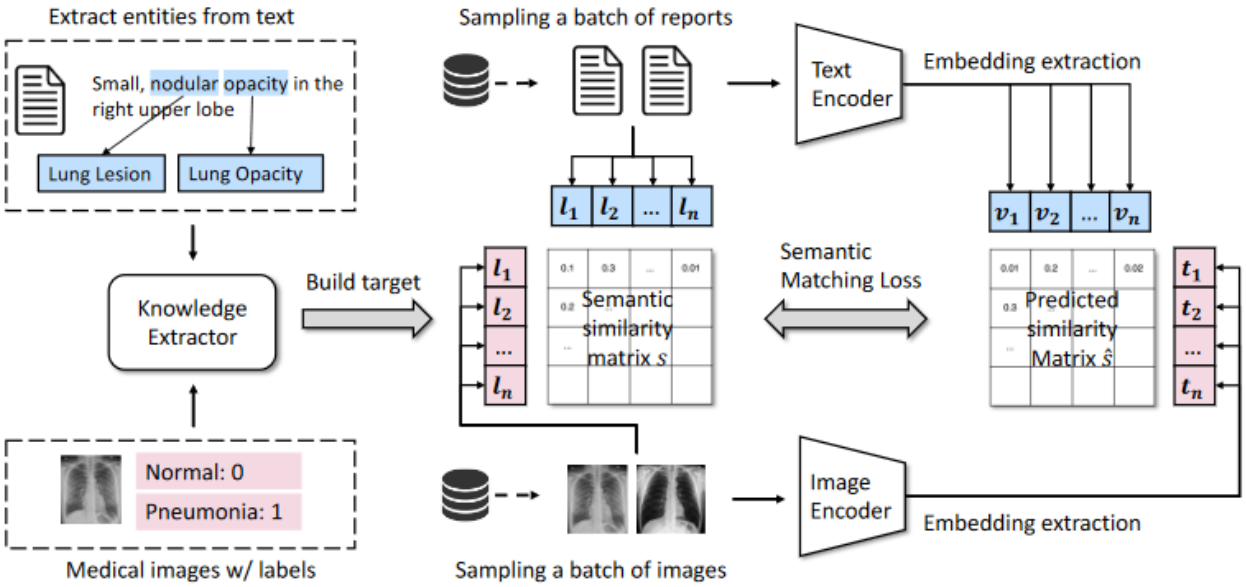


Figure 1. The MedCLIP Workflow. Note that the MedClip workflow is very similar to the basic CLIP workflow.

1. Introduction

1.1. Motivation

Within the past few years, the medical field has seen a significant increase in artificial intelligence applications, with large tech giants such as Google and Amazon racing to create their AI-powered healthcare solutions. Early this year, Google introduced Med-PALM2, their large language model (LLM), which passed an exam-style questionnaire that assesses a physician's ability to apply knowledge concepts, the US Medical Licensing Examination (USMLE) [4]. With such a race going on, the benefits of creating AI solutions that can help benefit healthcare professionals from diagnostics to treatments and diagnosis.

Whether or not Med-PALM2 will stand the test of time will soon be tested; however, as the field moves towards multimodal transformer-based LLMs, we will see more classic feature representations competing against contemporary techniques that newer models incorporate. One contemporary feature extraction technique like CLIP can enhance traditional domain-specific methodologies to rival modern classification algorithms. In this project, we will focus on chest X-ray and skin data, which have been studied using computer vision for decades. We seek to answer, "What Will Stand the Test of Time?"

1.2. Background Material

Contrastive learning focuses on extracting meaningful representations by contrasting positive and negative pairs of instances. Vision-text contrastive learning like CLIP [3] aimed to match the paired image and caption embeddings while pushing others apart, which improved representation transferability and supported zero-shot prediction. However, MedCLIP, built on top of CLIP, replaces the Noise-Contrastive Estimation (InfoNCE) with semantic matching loss based on medical knowledge to eliminate false negatives in contrastive learning [5].

The knowledge extraction module carries out the extraction of medical entities from unprocessed medical reports. Next, a semantic similarity matrix is generated by comparing medical entities found in text with the original labels extracted from images as shown in figure 1. This matrix allows for the pairing of any two independently selected images and text. Finally, the extracted embeddings from images and text are paired to align with the semantic similarity matrix [5].

In doing so, MedCLIP outperformed CLIP and another efficient labeling for medical image recognition framework, GLoRIA [2], on zero-shot prediction. With such claims, MedCLIP is a viable candidate for testing to see if domain-specific knowledge helps achieving better performance than traditional classification approaches.

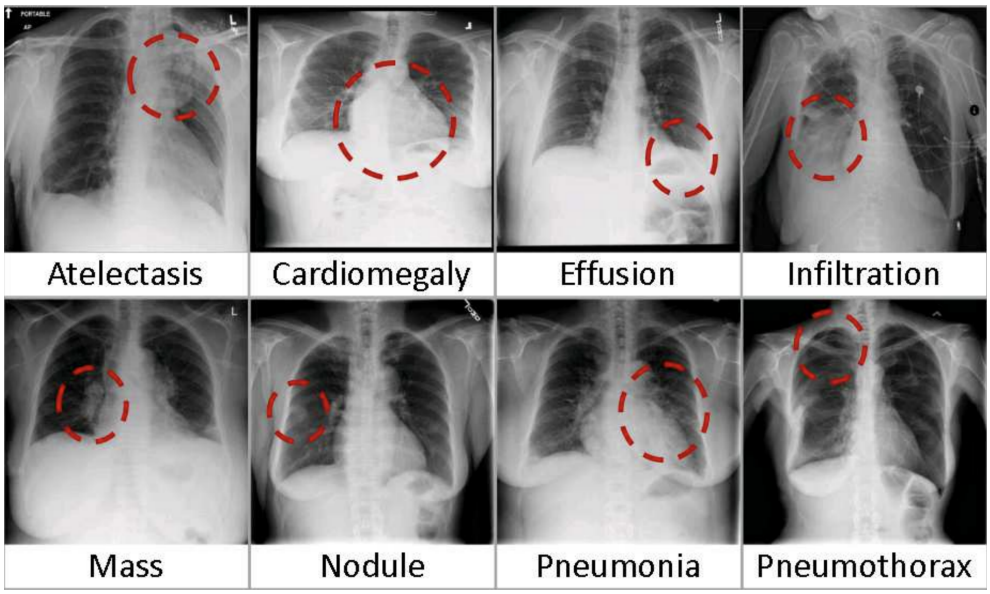


Figure 2. The figure shows labeled images from the NIH Chest X-ray dataset. The red circles indicate the region which indicates to doctors the presence of the disease.

1.3. Problem

We seek to determine whether using CLIP features can help traditional classification approaches achieving similar performance to modern systems. We are interested in medical images, which are challenging because they require a high degree of precision, and they are very different from the images in popular datasets used to train models like CLIP. As a result, we will also be testing MedCLIP as a ”counterpart” to the more generalized CLIP model, tailored specifically for the unique challenges posed by medical image analysis within the context of the Med-PALM2 project.

1.4. Datasets

Neither dataset we used was split into train and test sets, so we manually did a random split into two sets: a training set with 80% of the data and a test set with the remaining 20%.

HAM10000

The HAM10000 dataset was created to provide data for diagnosing pigmented skin lesions. The dataset has 10015 dermatoscopic images from different populations. The image labels included seven diseases, and diagnoses were confirmed through follow-up, expert consensus, or confirmation by in-vivo confocal microscopy. [1]

We use HAM10000 because skin lesion images differ significantly from those used to train MedCLIP and CLIP. As a result, we can better evaluate how each algorithm generalizes to new domains.

NIH Chest X-Ray

The NIH Chest X-ray dataset is a commonly used chest X-ray dataset for diagnosing lung diseases. The dataset has over 100,000 images with several diseases. Labels were determined similarly to the HAM10000 data. The labels include:

- Atelectasis
- Consolidation
- Infiltration
- Pneumothorax
- Edema
- Emphysema
- Fibrosis
- Effusion
- Pneumonia
- Pleural thickening
- Cardiomegaly
- Nodule
- Mass
- Hernia
- No Finding

[1] Note that some readers may find some of the HAM1000 images disturbing so they are not shown here.

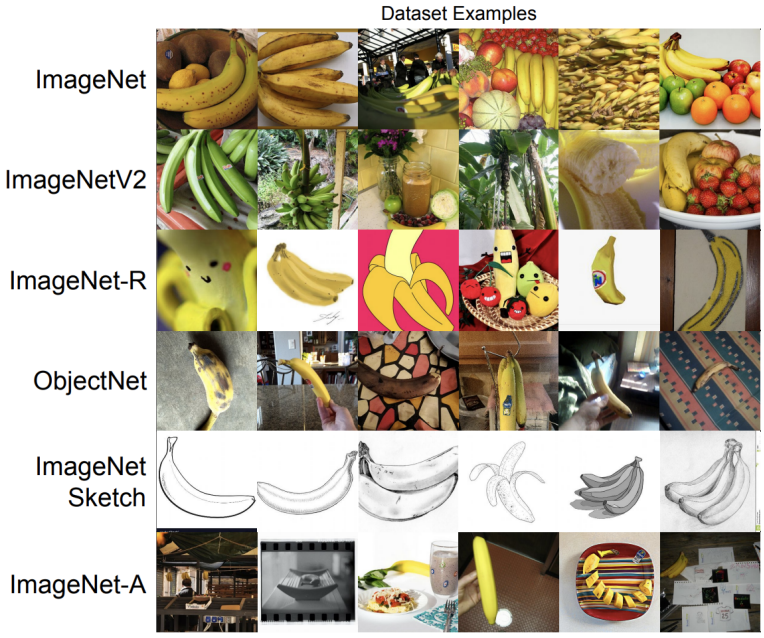


Figure 3. Examples of images that CLIP was trained on. They are very different from the medical images explored in this project.

1.5. CLIP

CLIP, developed by OpenAI, bridges the gap between natural language processing and computer vision. CLIP is trained on a large-scale dataset consisting of images paired with textual descriptions, many of which come from the ImageNet dataset [1]**??**. This training enables CLIP to understand and interpret images in the context of natural language. CLIP's architecture comprises two primary components: a vision encoder and a text encoder. The vision encoder processes images, transforming them into visual feature vectors, while the text encoder similarly converts textual descriptions into text feature vectors. These encoders are trained to align the image and text vectors in a shared high-dimensional space.

The training of CLIP uses contrastive learning, where the model is trained to match the correct pairs of images and texts while distinguishing them from incorrect pairs. This is achievingd by maximizing the similarity of vectors from matching image-text pairs and minimizing it for non-matching pairs. CLIP has been found to work well in zero-shot settings. This is because the model learns a broad representation of visual and textual information, enabling it to understand and classify images based on various textual descriptions. The model can be applied to a diverse range of domains, including but not limited to image classification, object detection, and tasks that require understanding nuanced visual concepts expressed in natural language.

1.6. MedCLIP

MedCLIP was trained on multiple chest X-ray datasets. The CheXpert dataset of chest X-rays with 14 observation labels was collected from Stanford Hospital, the MIMIC-CXR database from Beth Israel Deaconess Medical Center in Boston, the publicly available COVID dataset and the RSNA Pneumonia dataset from the National Institutes of Health (NIH). Examples of the data can be found in figure 4.

Accordingly, we tested on the NIH dataset because it is in distribution for MedCLIP. Additionally, lung disease diagnosis is a simple task for a human radiologist, so we hope that classifiers can achieving higher accuracy with better features.

1.7. Feature Learning

In the dynamic realm of AI, the evolution from manual feature selection to learned feature extraction marked a pivotal shift in how we approach machine learning, particularly in image recognition. Historically, the focus was predominantly on feature selection, a meticulous process where experts identified and handpicked features for image analysis. While effective in its time, this method was inherently limited by the scope of human foresight and the complexity of natural images.

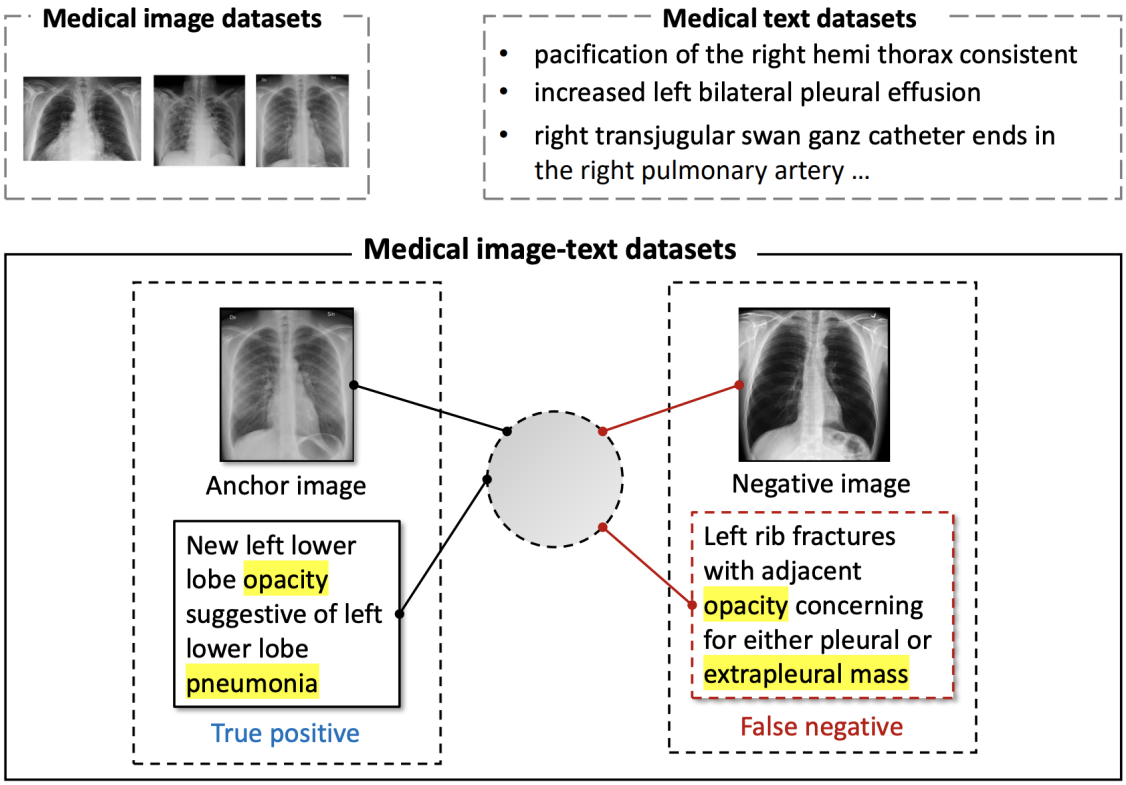


Figure 4. Illustration of why it is difficult to do contrastive learning on medical image data.

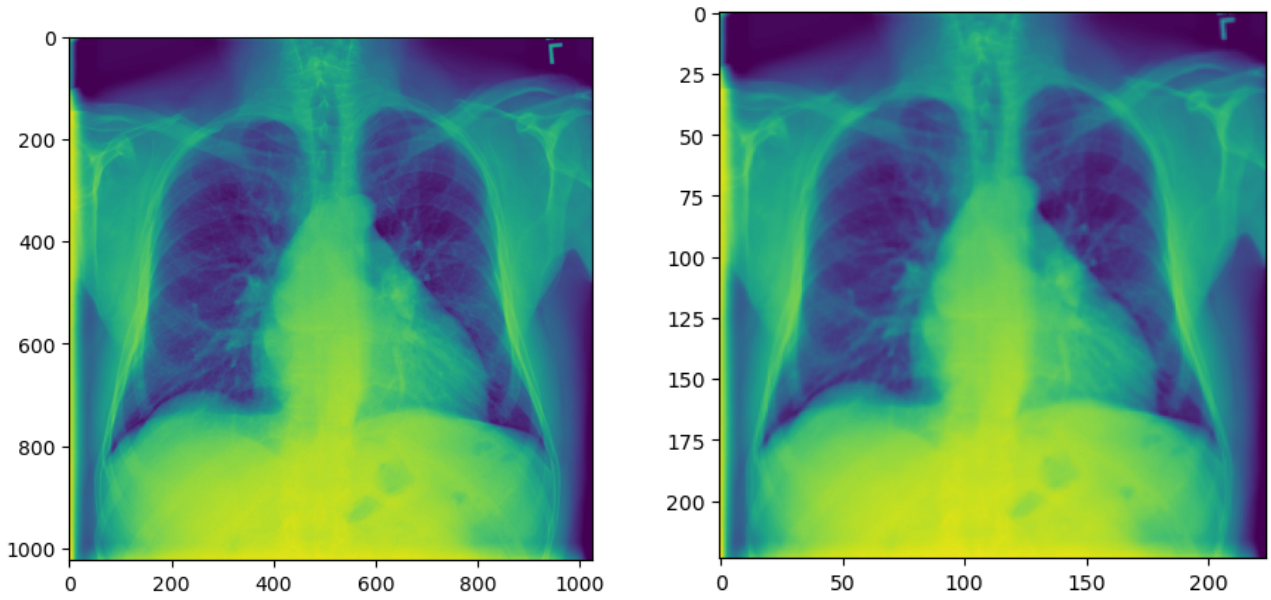
Enter CLIP (Contrastive Language–Image Pretraining), an innovative approach developed by OpenAI that harnesses the power of feature learning. CLIP revolutionizes how we interact with visual data by learning features directly from the data, using a vast array of images and text from the internet. This shift from manual feature selection to automatic feature learning is akin to teaching the system to understand and interpret visual information in a manner that's more aligned with how humans perceive and contextualize images from the vision-text contrastive learning model. CLIP developed a more nuanced understanding of visual content by analyzing and understanding images alongside their corresponding text descriptions. This method enables CLIP to recognize a wide range of objects and scenes with remarkable accuracy, far surpassing the limitations of traditional feature selection methods.

2. Approach

2.1. Data Preparation

The HAM10000 was split into 80% train and 20% while NIH was split into 90% train and 10% testing to get as much training for CLIP as possible. Even though CLIP is more of a generalized model, we wanted to see if it can compete with MedCLIP, given that MedCLIP is trained in the medical domain, particularly chest X-ray.

For CLIP, we transformed all images using CLIP's preprocess, a torchvision transform that converts a PIL image into a tensor, that was used for testing zero-shot capabilities, logistic regression, SVM, k-means clustering, and random forest classification for both HAM1000 and NIH dataset.



(a) Before transformation for NIH

(b) After transformation for NIH

Figure 5. Comparative view of NIH images before and after transformation

For MedCLIP, it was different since the authors of MedCLIP specifically have a preprocess that includes resizing all images to be 224 x 224, image means, and image standard deviation values that were used from their training and testing, which we used to get their results as close as possible. We used these transformations for all MedCLIP ResNet50 and ViT models' tests. See figure 5.

2.2. Base Models

Linear Probe

In our project, we incorporated logistic regression as a linear probe for the classification task, leveraging the robust and widely used scikit-learn library for its implementation. Logistic regression, a linear model, is particularly well-suited for binary classification problems but can be extended to multi-class classification. This aligns with our objective of classifying medical images into various disease categories.

We applied logistic regression to the extracted features obtained from the CLIP model. This approach enabled us to map these high-dimensional features onto a lower-dimensional space conducive for classification. As a very lightweight model, logistic regression demonstrates the runtime improvements offered by superior feature representations.

Support Vector Machine

Alongside logistic regression, we also employed Support Vector Machines (SVMs) as a part of our comparative analysis, utilizing the scikit-learn library for its implementation. SVM can handle both linear and non-linear classification tasks. It works on the principle of finding the hyperplane that best separates different classes in a high-dimensional space, maximizing the margin between data points of different classes.

K-Means

The sole unsupervised method we tested, K-means cluster samples, aimed to place each sample in the cluster with the closest mean. Note that we also tried K-Nearest Neighbors, which we implemented manually. However, due to its low performance and the fact that, historically, K-Means was preferred over K-Nearest Neighbors, we did not include it in our report. K-Means was implemented using scikit-learn as well. Because the algorithm is unsupervised, we set the number of clusters to the number of classes, expecting the model to learn to cluster by class.

Random Forest

We also analyzed the performance of Random Forests. Decision trees have been historically popular for diagnosis because of

their interpretability. As one of the highest-performing approaches in the 1990s and 2000s, random forests were an obvious choice to evaluate feature learning methods.

Random forests were implemented with scikit-learn. The forest consisted of 100 trees, which used the Gini Impurity to measure split quality. We did not set a maximum tree depth.

ResNet

In our project, we also integrated ResNet50, a deep convolutional neural network, as a modern benchmark for comparison against our hybrid models. ResNet50, short for Residual Network with 50 layers, is a variant of the ResNet model that uses residual connections. These connections help mitigate the vanishing gradient problem commonly encountered in deep networks, thus allowing for effective training of much deeper networks. ResNet is a popular and modern classification method that will help determine whether the older methods can withstand time when combined with modern learned features.

We evaluated both zero-shot and fine-tuned ResNet50. In each case, we started by loading the pre-trained model weights from PyTorch. We used the IMAGENET1K_V2 and the corresponding image transforms. Next, we replace the last fully connected layer with a smaller layer. ResNet50 was trained on ImageNet, which has many more classes, so we needed the network output size to match the number of classes depending on which dataset we were using.

In order to fine-tune ResNet50, we started from the pre-trained model and tuned the entire model for 50 epochs, with a learning rate of 10^{-5}. The success with a large number of epochs and relatively large learning rate is likely due to the fact that ResNet50 was not trained on medical images, so the pre-trained model was not well suited to medical image classification.

2.3. Feature Learning

In the 1990s and early 2000s, state-of-the-art approaches used the base models we tested and other relatively simple models compared to deep networks. Consequently, selecting good features was critical to achieving high performance at the time. Hence, we apply modern features extracted by CLIP and MedCLIP to use with the classical methods.

In order to extract features, we provide a `get_features` function, which takes a dataloader and feature extraction model (either CLIP or MedCLIP), and returns the learned features. The CLIP models work by creating image and text embeddings for an image-text pair. They are trained with contrastive learning, so related image-text pairs have more similar embeddings.

As an additional method for comparison, we used zero-shot classification with each of the CLIP methods. We start by storing text embeddings for each class label to do zero-shot classification with these models. Then, we compute each image's embedding and find the most similar class label by comparing the Euclidean distance between embedding vectors.

In order to do classification, we use just the image embedding, using the embedding as the model input rather than the raw image, for both training and inference. We evaluate the results of each method three times: using the raw images as features, using CLIP features, and using MedCLIP features.

3. Results

As previously mentioned, to help with our experiments, we designed a `get_features` method which extracts features from images. The extraction method is either an identity function, to return the original image, the CLIP model, MedCLIP-ResNet50, or MedCLIP-ViT. We wrote a notebook to evaluate data on each model: Zero-shot (for feature learning), linear probe, SVM, K-Means, Random Forest, and ResNet50. Then, to get results, we choose a dataset to load and a feature extraction method and run the notebook. We evaluate on a hold-out test set, computing each model's accuracy.

In table 1, we see how feature learning impacts performance. We see that CLIP and MedCLIP perform similarly, which is expected since neither is trained on images like those found in the HAM1000 dataset. In this case, the linear probe with CLIP features performed almost as well as ResNet, suggesting that CLIP may in fact be able to keep linear classifiers in the competition for state-of-the-art models.

The data in table 2 is somewhat surprising. MedCLIP generally did not perform as well as CLIP when generating features despite being trained on similar images.

We also evaluated the zero-shot performance of each technique, as shown in table 3. The results show MedCLIP outperforming CLIP, unlike in the settings where it was used with another model.

4. Conclusion

Overall, the results differed significantly from our expectations. We expected to see a minor boost from CLIP, with a more significant boost from MedCLIP, especially for the NIH Chest X-ray data. What we found was that using learned feature

Features	Linear Probe	SVM	K-Means	Random Forest	ResNet50
Pixel Values	N/A	N/A	N/A	N/A	85.6
CLIP	**56.975**	**54.245**	40.885	**55.342**	N/A
MedCLIP-ResNet50	54.995	33.518	41.332	53.951	N/A
MedCLIP-ViT	56.894	34.748	**42.133**	54.772	N/A

Table 2. The accuracy on **NIH** Chest X-Ray test set for each model using each set of features. The CLIP features are the embeddings computed from each of the CLIP models.

Model	HAM10000	NIH Chest X-ray
CLIP	21.147	0.572
MedCLIP-ResNet50	22.346	**53.138**
MedCLIP-ViT	**27.593**	16.531

Table 3. The zero-shot accuracy using each set of features.

Model	Zero-Shot	Linear Probe	SVM	K-Means	Random Forest	ResNet50
CLIP	7.833	10.815	36.209	117.76	0.778	N/A
MedCLIP-ResNet50	10.58	22.37	56.209	N/A	N/A	N/A
MedCLIP-ViT	216.81	22.59	66.209	N/A	N/A	N/A

Table 4. Runtimes of the NIH dataset converted from seconds to minutes

extractors yielded a clear performance improvement. In some cases, it was good enough to justify using the older method with learned features over ResNet50 for some use cases, but for the most part, the methods still lagged behind the more modern models. An interesting area for future exploration would be to use CLIP and MedCLIP features with smaller neural networks and compare them with more extensive networks like ResNet.

The feature extraction methods likely did not work as well for the HAM1000 dataset because the dataset's images are too dissimilar from the models' training data. Furthermore, MedCLIP may not have performed as well as CLIP on HAM10000 because it was only trained on one type of image, so it does not generalize as well as CLIP.

It is unclear why MedCLIP did not consistently outperform CLIP on the NIH Chest X-ray data, given that MedCLIP was trained exclusively on X-ray data. However, one possible reason is that it was designed to encode long diagnoses rather than short labels, so once again, CLIP's ability to generalize helps it beat MedCLIP. Additionally, MedCLIP expects bounding boxes to show where the disease would be. We did not provide those bounding boxes, so that it would not have more information than CLIP. The lack of bounding boxes caused a minor decrease in the zero-shot results compared to what was reported in the original paper.

The zero-shot results, on the other hand, were more in line with our expectations. They showed both MedCLIP models beating CLIP on both datasets. The discrepancy between our experiments may result from the models we evaluated. This suggests we either try new models or adjust the training. Regardless, the results suggest that on its own, MedCLIP is a superior model for medical images.

Despite the unexpected results, these feature extraction methods hold exciting potential to help older models stand the test of time against larger models. Furthermore, while their performance was worse than modern state-of-the-art models, their performance was high enough that developers may wish to choose them if they are willing to sacrifice some accuracy in favor of a faster model.

5. Statement of individual contribution

- **Joshua:** Joshua implemented pipelines for each of the classifiers: K-Nearest Neighbors, Random Forests, SVM, ResNet50, and linear probe. His pipelines included functions to load datasets and compute features from models like CLIP and MedCLIP. Joshua implemented the HAM10000 dataset in Python and fine-tuned ResNet50 on HAM10000 and the NIH dataset. Josh also executed code to evaluate the methods and evaluated the performance of CLIP and ResNet on HAM10000. Joshua wrote a significant portion of the report as well.

- **Mario:** Mario implemented MedCLIP's ResNet50 and ViT models to use Josh's classifiers pipelines and HAM10000 dataset. Mario implemented the NIH dataset along with its 15 classifiers to be used for CLIP, MedCLIP ResNet50, and MedCLIP ViT models. Mario also executed code for several experiments, including all experiments with either MedCLIP or the NIH Chest X-ray dataset. He also collected the runtime results. Finally, Mario contributed to the results section of the report and wrote most of the introduction.

6. ChatGPT Documentation

We used ChatGPT to help us write about feature learning, and why clip is helpful including that historically, feature selection was the main focus, so now we can use learned features instead. Full prompt and response located here with this ChatGPT conversation.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 4

[2] Shih-Cheng Huang, Liyue Shen, Matthew P. Lungren, and Serena Yeung. Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3922–3931, 2021. 2

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 2

[4] 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, 2023. 2

[5] Zifeng Wang, Zhenbang Wu, Dinesh Agarwal, and Jimeng Sun. Medclip: Contrastive learning from unpaired medical images and text, 2022. 2