Improving Model Transparency: Transferability and SHAP Analysis for Deep Neural Network Debugging

Rebecca Tsekanovskiy[1] and Neha Keshan[1]

Rensselaer Polytechnic Institute NY 12180, USA tsekar@rpi.edu, keshan2@rpi.edu

Abstract. In the last couple of years, the need to understand deep learning models, particularly black-box algorithms, has grown significantly, amplifying the demand for transparency and interpretability. Explainable Artificial Intelligence (XAI) techniques have been evolving to address these issues by providing clear insights into the model behavior. This paper explores the effectiveness of SHapley Additive exPlanations (SHAP) in debugging deep learning neural networks. We review traditional debugging methods and compare their limitations to the advanced capabilities of SHAP. Our methodology involves training a deep convolutional neural network (DCNN) using thousands of medical images, followed by SHAP analysis to provide detailed explanations of the model's predictions. The results demonstrate high accuracy and low loss on both training and validation datasets, indicating the model's robustness. SHAP visualizations confirm that the model's decision-making process is transparent and focused on relevant features. This study highlights the limitations of traditional debugging methods and showcases the practical insights and critical role of XAI techniques in developing interpretable and reliable AI models.

Keywords: Black Box Algorithms · Deep Learning · SHAP (SHapley Additive exPlanations) · XAI (Explainable Artificial Intelligence) · Convolutional Neural Networks (CNN)

1 Introduction

1.1 Motivation

The rapid advancements in deep learning have led to its widespread adoption across various industries. However, the inherent complexity of deep learning models, particularly deep convolutional neural networks (DCNNs), poses significant challenges in understanding and interpreting their decision-making processes. This opacity is often referred to as the "black box" problem, where the internal workings of the model remain obscure, even to domain experts.

This paper aims to explore the effectiveness of SHAP in debugging deep learning neural networks. By comparing traditional debugging methods with SHAP, we demonstrate how SHAP can improve model interpretability, identify data quality issues, and ultimately enhance model accuracy and reliability. Our study contributes to the growing body of research on XAI by providing practical insights into the application of SHAP in real-world scenarios, paving the way for more transparent and accountable AI models.

1.2 Research Contribution

Despite the significant advancements in deep learning, DNNs, there remains a substantial challenge in effectively debugging and interpreting these models. Traditional debugging methods for DNNs often lack transparency, making it difficult to understand why a model makes certain predictions. This is especially problematic in applications requiring high reliability and trust, such as medical image analysis, where incorrect predictions can have severe consequences. Moreover, existing methods to enhance model interpretability, such as SHapley Additive exPlanations (SHAP), are not fully optimized or thoroughly evaluated in the context of deep neural networks, leaving a gap in understanding the model. We evaluated a DNN trained on lung scans and tested it on brain scans without retraining, finding that the model fails to generalize across domains, as evidenced by SHAP analysis highlighting irrelevant features and significantly lower performance metrics on brain scans. Within the lung scan domain, the SHAP analysis revealed inconsistent feature focus and potential overfitting. These findings underscore the need for domain-specific training and robust interpretability tools. Our contributions include demonstrating the limitations of cross-domain generalization in DNNs, identifying specific areas for improvement in model training and debugging, and validating the effectiveness of SHAP for interpreting medical image models.

1.3 Specific Focus and Relevance

This research specifically focuses on the transferability of features learned by a DCNN model trained one one type of cancer (normal, malignant, benign) when using that model on an entirely different type of cancer scan. By using SHAP, we aim to provide a trasparent understanding of how features transfer across diferent types of cancers. This is important because of the following:

1. Understanding features that are specific to a certain type of cancer can help with fine tuning models focusing on a specific disease.
2. Understanding features that are universal to cancer can help with developing general models
3. Enhancing a models adaptability to different types of data, specifically in healthcare where variability in data occurs.

1.4 Deep learning

Deep learning (DL), a type of machine learning (ML), has changed various industries by causing models to learn and make decisions without the need for humans. This consists of using neural networks, which is made up of multiple layers of interconnected nodes, helping to understand and make sense of the data. [19] This is done in a way that simulates the functionality of neurons, ultimately contributing to the concept of neural networks [4]. These neural networks are created in a way similar to how a human brain process information[4]. The concept of deep learning was proposed as an artificial neural network(ANN) model with several different layers [19]. The average DL model consists of different layers: input layer, multiple hidden layers, and an output layer. Data will go through each layer and have information extracted from it and passed onto the next layer [16, 19].

DL models can be classified into four categories: deep supervised, unsupervised, reinforcement learning, and hybrid models [19]. Deep supervised learning models show their strength in tasks such as natural language processing, image recognition, and predictive analytics, whereas a regular algorithm falls short due to their inability to handle unstructured and large datasets[16, 19]. In the aspect of our project, we will be focusing on deep convolution neural network, a type of deep learning used for image recognition [18].

Deep Neural Networks (DNNs), a subcategory of deep supervised learning, are only becoming more intricate [19]. With the large numbers of layers, their decision making processes make it difficult to understand how and why specific predictions were made. Despite this, DNNs outperform that of a traditional ML model, yielding results with high accuracy. [4]

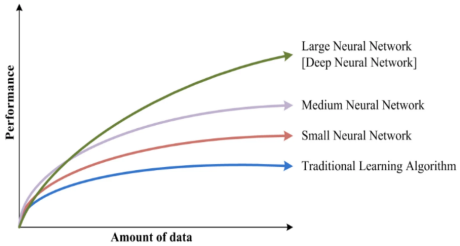

Fig. 1. Amount of data vs. Performance
[4]

Black Box Algorithms Deep learning models, specifically DNNs, are sometimes mentioned as a "black box" algorithms because of their complicated and hidden decision making processes [15]. In other words, it is unclear what information causes a model to come to a certain conclusion. With this, many applications view this as a huge negative. This is causing a strain on a majority of users wanting to trust and adopt these models, like in stake areas like healthcare and finance [19, 15]. In some scenarios, the reasoning behind a certain conclusion may not be critical, but in cases like self-driving cars, understanding it is crucial due to the potential impact on human lives. [15]. Today, these models are being trained with millions of different datasets, causing the model to observe certain patterns that a human may not have been able to identify [15]. Thus, there has been a growing need to understand black box algorithms, creating the demand for explainable Artificial Intelligence (XAI).

Explainable Artificial Intelligence and SHapley Additive exPlanations Explainable Artificial Intelligence techniques have risen to help combat the transparency issues with black box algorithms. The goal of XAI is to ensure there is interpretabilty privacy, robustness, trust, fairness, and transparency. SHapley Additive exPlanations (SHAP) is a type of XAI technique based off a cooperative game. SHAP helps a user understand the contributions of different features to a model's predictions by quantifying all aspects [20]. SHAP is also possible of both global and local explainability[6]. This means it can provide detailed insights into individual predictions as well as an understanding of the overall behavior of the model across the entire dataset.[6] This dual capability makes SHAP a powerful tool for enhancing model interpretability and making sure AI systems are both transparent and accountable.

By using SHAP to help explain deep learning models, we can better understand the underlying framework of these systems, identify data quality issues, and improve model accuracy. This study explores the effectiveness of SHAP in debugging deep learning neural networks, paving a new path for the future by developing more transparent AI models.

2 Literature Review

This section reviews various methodologies and approaches to interpreting and explaining deep learning models. We focus on sensitivity analysis, layer-wise relevance propagation, Locally-interpretable model-agnostic explanations, and SHapley Additive exPlanations. Additionally, we explore the application and evaluation of SHAP in different domains.

2.1 Interpreting and Explaining Deep Learning Models

There are many different methods in place to help understand a deep learning model. This includes sensitivity analysis, layer-wise relevance propagation, SHapley Additive exPlanations, Locally-interpretable model-agnostic explanation, and more [2, 15].

Sensitivity analysis (SA) refers to explaining a models prediction from the locally evaluated gradient [15]. For each input variable, an image pixel, sensitivity analysis quantifies each input. Here, it is mathematically represented as [15]

$$R_i = ||\frac{\partial}{\partial x_i} f(x)||. \tag{1}$$

The system will first identify the input image and classify it. Then, a heatmap will be able to be visualized, demonstrating each pixels worth towards the prediction [15]. It is identifying the pixels needed to be altered to make the image resemble more of the prediction. A change in these pixels would ultimately impact classification score [15]. With this, there are drawbacks, such as SA in explainability often focuses on irrelevant features, leading to misleading heatmaps and poor quantitative performance in accurately identifying key predictive features [15].

Layer-Wise Relevance Propagation (LRP) can also help explain predictions made by models by showing which specific input is crucial for the prediction[15]. LRP can provide a detailed breakdown on how each part of the input impacts the output. LRP can also be used with different AI models, not specifically just deep neural networks [15]. The effectiveness of LRP can be sensitive to the chosen parameters. Incorrect parameter settings can affect the accuracy and reliability of the relevance scores [15]. Moreso, LRP can be computationally expensive, especially in cases when the model is large [15].

Locally-interpretable model-agnostic explanations (LIME) is used as an XAI technique to help simplify the interpretation process by approximating the model with a linear model [13, 21]. In other words, the goal of LIME is to represent the behavior of a complex model, such as the black box model, in the realm of a specific data instance [13]. LIME can be applied to any machine learning model because it is model-agnostic. LIME can provide information on why the model came to a certain prediction, more specifically on the local level[21]. LIME does not try to approximate the black box model on the global level because it may not always be feasible due to the complexity[13].

SHAP has stemmed from a cooperative game theory, used to determine the impact certain features have on a model's prediction, and it is not impacted by complexity or structure of the model. This is because each feature is assessed based off all possible feature combinations. SHAP contains many different great qualities, ultimately contributing to the popularity. SHAP can be applied to decision trees, neural networks, linear models, and more [20]. Since it can evaluate both on the global and local level, unlike most other popular XAI tools, it makes it extremely versatile. The usage of SHAP values have been used for helping transparency and improving model debugging [20].

SHAP calculates the average output of the model when only specific features are identified. This can help understand what the model predicts when information is limited [20]. SHAP also calculates the contribution of an individual feature by focusing on the change of the model's prediction, specifically in cases where a new feature is added[20]. Ultimately, this is demonstrating how much weight a certain feature carries in relation to a combined subset of other features. SHAP will also calculate the value for a feature by averaging the way it is contributing across all possible combinations to order the features, making sure that the SHAP value accounts for every possible variation providing an accurate measurement[20]. Moreso, if a complex model is made from several simpler models, the SHAP value for a feature in the complex model is the sum of the

SHAP values for that feature in each simpler model, adjusted by their weights. It must be noted that SHAP explanations are calculated in #P-hard, meaning it is extremely computationally intense and can not always compute SHAP values in polynomial time [20]. Because of this, the practical application of SHAP can be limited for large models.

2.2 Application and Evaluation of SHAP in Various Domains

SHAP has been implemented in multiple different domains, demonstrating its effectiveness in various aspects.

Healthcare SHAP has been applied in the healthcare sector, ultimately aiding in combining transparency and AI. In one aspect, the authors of [9] implemented eXtreme Gradient Boosting (XGBoost), a type of decision tree, with SHAP to analyze predictions for heart failure stages. Using SHAP, researchers were able to view how clinical features impacted the model's prediction. For instance, the study identified gender, BMI, and blood pressure as significant predictors of heart failure stages, validating the model's logic and highlighting areas for potential refinement[9]. Moreover, it was stated that "in the SHAP interpretation for our prediction model, heart failure patients tend to have EF scores higher than average if the BMI value is high and vice versa"[9]. Using SHAP has helped better debugging, improving the model's overall accuracy and reliability.

By using SHAP, the researchers were able to find information that they would not have been able to recognize by just analyzing the information given by XGBoost[9]. Additionally, the article highlights how SHAP can be used to detect and mitigate biases in deep neural networks. The researchers identified gender differences in the model's predictions, which could indicate potential biases in how the model processes clinical data[9]. The researchers conclude they believe it is possible for future use of machine learning models in the health sector to aid with clinical trials[9].

While this specific example utilizes XGBoost rather than deep neural networks, the principles of using SHAP for model interpretability and debugging are universally applicable across different types of models. The ability of SHAP to provide detailed feature contributions and enhance transparency can similarly benefit deep neural networks, specifically in critical applications.

3 Methods and Data

We first began by gathering a dataset of over 10,000 total images containing of abdomen and breast scans. Each image has a size of 66x66 pixels. The dataset was split into training, validation, and test in a 70:20:10 ratio. We are creating a dcnn in our work by training it on a diverse dataset and evaluate its ability to accurately recognize and classify these images. We got our dataset from here [11]. We used some code from [10].

3.1 Initial Methods

We will be using a Juypter notebook for better organization during data collection to perform the following:

1. Data Collection
2. Installations
3. Data Preprocessing
4. Data Loading
5. Architecture Design
6. Training
7. Evaluation
8. Deployment and Testing
9. Interpretation with SHAP

Installation, Pre Processing, Data Load We first began by installing all of our dependencies, and then checking if they are in a valid format (i.e: .jpg, .png). We use ‘TensorFlow‘ to help build and train our model. We use ‘OS‘ to help handle file paths. We use ‘cv2‘ to help deal with image recognition tasks. We use ‘imghdr‘ to help with the verification of data type. We use ‘pylot‘ to help generate graph visualizations.

After understanding the model was able to quickly identify the difference between the scans, we turned to using a more niche dataset where the scans closely resemble eachother.

3.2 Chest and Stomach Scans

We will now show sample images from the dataset to demonstrate the types of images used for training and validation.

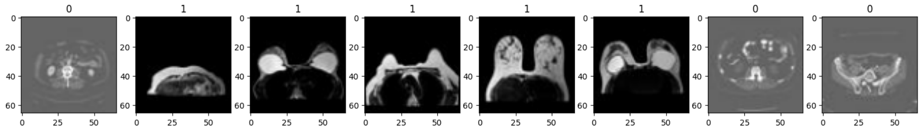

Fig. 2. Sample batch from the dataset, representing the two image classes

The images are labeled either as ’0’ or ’1’, which corresponds to the two classes the model is trained to distinguish between.The images depict various features that are characteristic of each class. In this visualization, the differences between classes are clear, showing the types of patterns the model is learning.

3.3 Lung Cancer Scans

We changed the model architecture for better accuracy on a dataset that closely resembles eachother. On a dataset size of approx 1,000, we had three different categories: benign lung cancer scans, malignant lung cancer scans, and normal lung scans.

```
model = Sequential([
    Conv2D(filters=64, kernel_size=(3,3), padding="same", activation="relu", input_shape= in
    Conv2D(filters=64, kernel_size=(3,3), padding="same", activation="relu", name="L2"),
    MaxPooling2D((2, 2)),

    Conv2D(filters=128, kernel_size=(3,3), padding="same", activation="relu", name="L3"),
    Conv2D(filters=128, kernel_size=(3,3), padding="same", activation="relu", name="L4"),
    MaxPooling2D((2, 2)),

    Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", name="L5"),
    Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", name="L6"),
    Conv2D(filters=256, kernel_size=(3,3), padding="same", activation="relu", name="L7"),
    MaxPooling2D((2, 2)),

    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name="L8"),
    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name="L9"),
    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name="L10"),
    MaxPooling2D((2, 2)),

    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name="L11"),
    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name="L12"),
    Conv2D(filters=512, kernel_size=(3,3), padding="same", activation="relu", name=’L13’),
    MaxPooling2D((2, 2)),

    Flatten(),
    Dense(256,activation = "relu"),
    Dense(64,activation = "relu"),
    Dense(class_count, activation = "softmax")
])

model.compile(Adamax(learning_rate= 0.0002), loss= ’categorical_crossentropy’, metrics= [’ac
```

3.4 Initial Results for Chest and Stomach Scans

The model was able to identify that it found 4990 files belonging to 2 classes. The model showed high performance during training and validation. After the fifth epoch, the model had the following statistics:

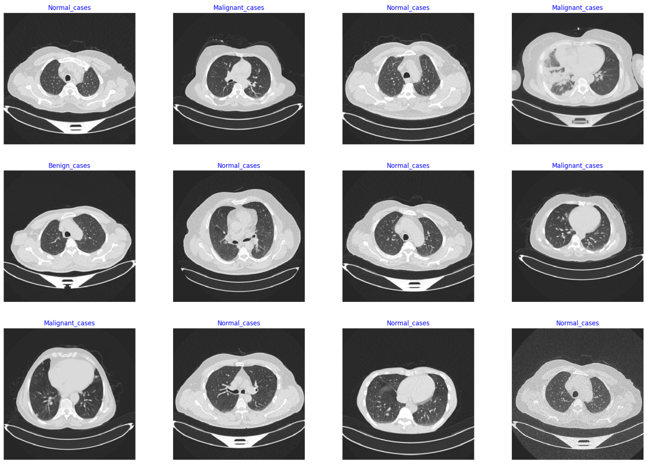

Fig. 3. Sample batch from the dataset, representing the three image classes

1. The model was trained for five epochs with the following results:
   - Epoch 1: Training accuracy: 98.37%, Validation accuracy: 100%
   - Epoch 2: Training accuracy: 98.60%, Validation accuracy: 100%
   - Epoch 3: Training accuracy: 100%, Validation accuracy: 100%
   - Epoch 4: Training accuracy: 100%, Validation accuracy: 100%
   - Epoch 5: Training accuracy: 100%, Validation accuracy: 100%

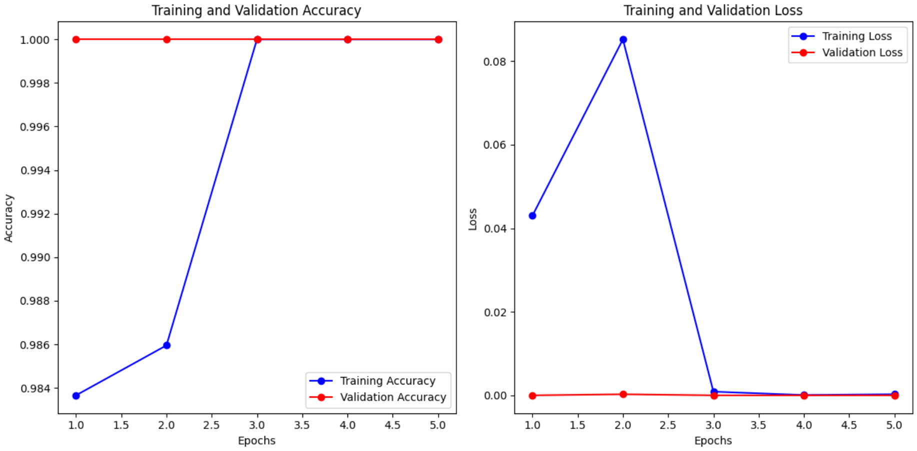

Fig. 4. Validation and accuracy as epoch's increase

The model achieved high accuracy on the test set, showing its effectiveness in distinguishing between the two classes. By the third epoch, accuracy hit 100%

and remained consistently at this level. The training loss is relatively higher in the first epoch and then drops significantly during subsequent epoch runs. By the second epoch, the training loss drops dramatically, nearing zero by the third epoch. This shows the high accuracy from the model performance for both learning and generalization given the near perfect accuracy scores. The training and validation curves are close, showing there was no over fitting done from the model. The low validation loss indicates that the model is generalizing well to images it has yet not seen. The SHAP values identified important features that the model is indeed using effectively, as reflected in the performance metrics.

Several visualizations were generated to illustrate the SHAP values:

1. Original Image: Displayed without any modifications.
2. SHAP Values for Class 0 and Class 1: Highlighted regions contributing to the predictions for each class.
3. Overlay of SHAP Values on Original Image: Combined SHAP values with the original image.
4. SHAP Values for Each Color Channel: Visualized SHAP values for Red, Green, and Blue channels separately.
5. Difference Between SHAP Values for Class 0 and Class 1: Highlighted areas where the model's decision-making differed between the two classes.

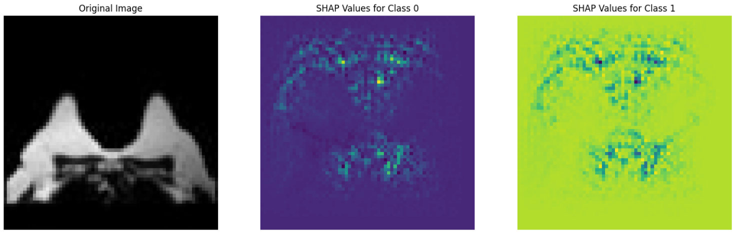

Fig. 5. SHAP values interpreted

The leftmost image is the original image from the dataset without any change. The middle image represents Class 0, showing the contribution each pixel had to the image. Brighter regions represent stronger influences towards Class 0. The rightmost image shows the SHAP values for Class 1. Like the previous image, bright regions show pixels that drive the model's prediction. This is extremely helpful because if the model is making incorrect prediction, in other words, the training accuracy for the model is low, it is possible to identify what specifically is causing the model to be confused. This is helping to make the model's decision making process more transparent and helpful. Ultimately, this is helping to improve debugging for multiple reasons.

1. Visualize SHAP values for a batch of correct and incorrect classified images and compare the SHAP value patterns.

2. Identify any common patterns in the SHAP values causing for the misclassification.
3. Adjust a multitude of parameters based off the insights, like changing the model architecture, retraining the model, increasing the number of epoch runs.

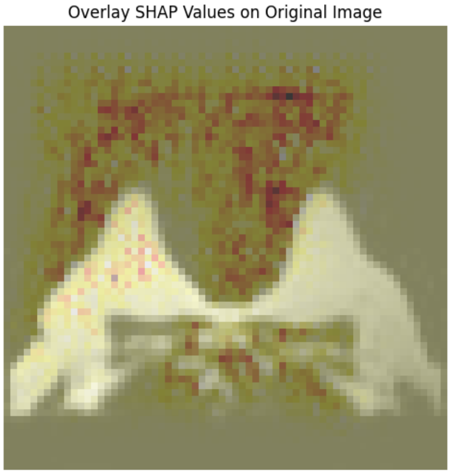

Fig. 6. An overlay of SHAP values on the original image, visualized through a heatmap

3.5 Initial Results on Lung Cancer Scans

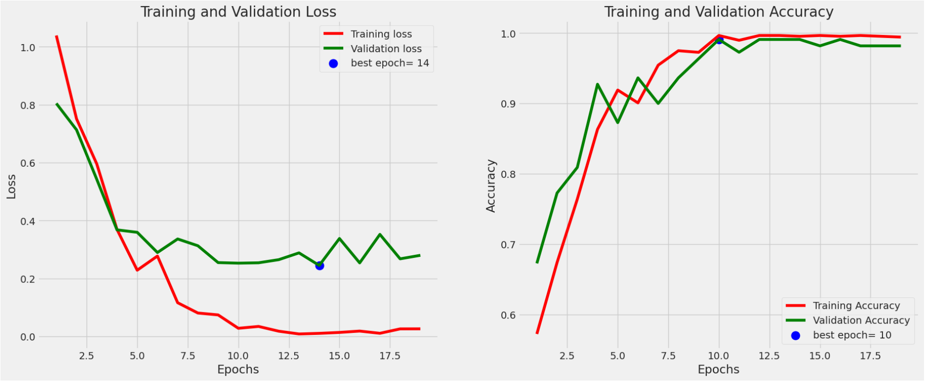

Fig. 7. Accuracy and training data

We were able to achieve a model and achieve 98.4375 validation and testing accuracy. Here is a confusion matrix showing what the model was focusing on when making decisions.

The SHAP values provide a comprehensive picture of how different parts of the image influence the model's predictions across all classes. Even if the scan is normal, the SHAP values for other classes can be significant, indicating the model's decision-making process involves ruling out other possibilities by highlighting what it does not see (evidence for benign or malignant) as much as what it does see (evidence for normal).

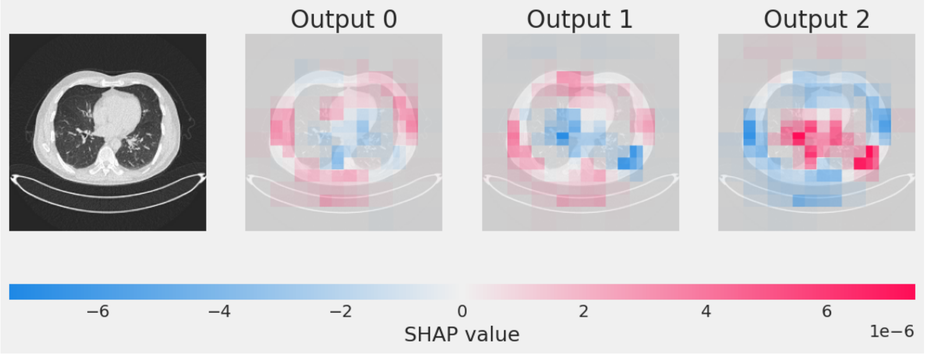

Fig. 8. Using a benign scan to see what part the model is focusing on to make the correct prediction.

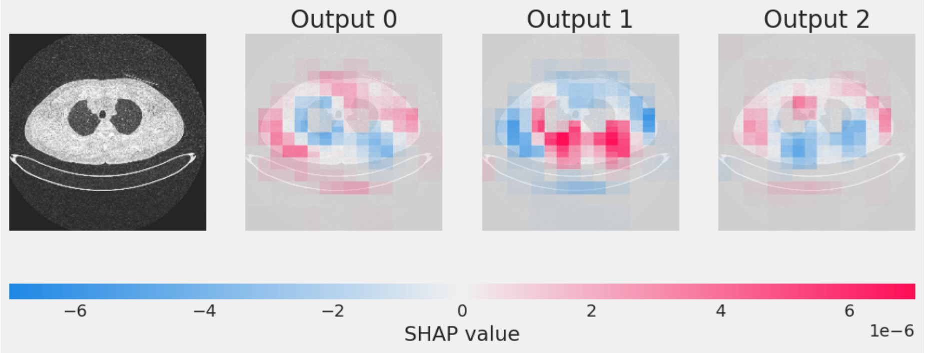

Fig. 9. Using a normal scan to see what part the model is focusing on to make the correct prediction.

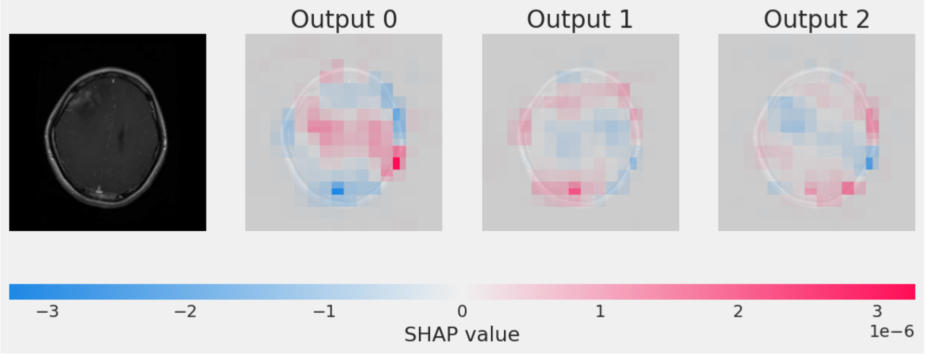

Fig. 10. Using a glioma scan to see what part the model is focusing on to make the correct prediction.

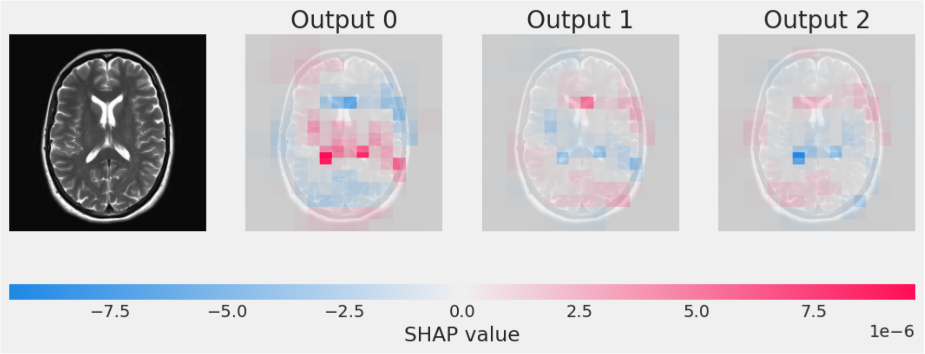

Fig. 11. Using a normal brain scan to see what part the model is focusing on to make the correct prediction.

3.6 Initial Results Analysis

Our study aimed to evaluate the generalization capabilities of a deep learning model trained on lung scans when tested on brain scans, without retraining. Additionally, we assessed the model's performance on lung scans to identify potential issues in generalization within its own domain. When applying the lung-trained model to brain scans, the SHAP analysis revealed several key observations:

1. The regions highlighted by SHAP values in the brain scans did not correspond to clinically relevant features for brain pathology. Instead, the model appeared to focus on irrelevant areas, indicating a misalignment in feature extraction.

These results suggest that the features learned from lung scans are not directly transferable to brain scans without domain-specific retraining. This finding demonstrate the necessity of training or fine-tuning models on data specific to the target application.

Generalization Within Lung Scans Certain regions highlighted by SHAP values did not align with known medical features of lung pathology, suggesting that the model might be focusing on irrelevant patterns in the training data. To address these issues, we plan to implement data augmentation and regularization techniques to enhance the model's robustness and generalization capabilities.Some scans, but not all, showed the model focusing on regions that are not typically associated with lung pathology, such as areas outside the lung field that are irrelevant to the diagnosis. Further validation will be conducted to ensure the model's focus aligns with clinically relevant features.The variability in SHAP values and the focus on irrelevant regions indicate potential overfitting. The model might be memorizing specific patterns in the training data that do not generalize well to new, unseen data.

Initial Results Concluded Our initial results demonstrate that the model, when trained on lung scans, does not generalize effectively to brain scans, highlighting the need for domain-specific training. Additionally, the SHAP analysis within lung scans suggests areas for improvement in the model's feature extraction and generalization processes. Future work will focus on refining the model to enhance its robustness and clinical applicability across different types of medical imaging data. Another possibility may include using pre-trained models on similar tasks and fine-tuning them on lung scans can help in learning more generalizable features.

References

1. Adadi, Amina, and Mohammed Berrada. 2018. "Peeking inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)." *IEEE Access* 6: 52138–60. https://doi.org/10.1109/access.2018.2870052.

2. Aso Bozorgpanah, and Vicenç Torra. 2024. “Explainable Machine Learning Models with Privacy.” *Progress in Artificial Intelligence* 13 (1): 31–50. https://doi.org/10.1007/s13748-024-00315-2.
3. Barreno, Marco, Blaine Nelson, Anthony D. Joseph, and J. D. Tygar. 2010. “The Security of Machine Learning.” *Machine Learning* 81 (2): 121–48. https://doi.org/10.1007/s10994-010-5188-5.
4. Bhatt, Chandradeep, Indrajeet Kumar, V. Vijayakumar, Kamred Udham Singh, and Abhishek Kumar. 2020. “The State of the Art of Deep Learning Models in Medical Science and Their Challenges.” *Multimedia Systems*, September. https://doi.org/10.1007/s00530-020-00694-1.
5. Clement, Tobias, Hung Truong, Nils Kemmerzell, Mohamed Abdelaal Abdelaal, and Davor Stjelja. 2024. “Beyond Explaining: XAI-Based Adaptive Learning with SHAP Clustering for Energy Consumption Prediction.” Ar5iv. February 2024. https://ar5iv.labs.arxiv.org/html/2402.04982.
6. Kostopoulos, Nikos , Dimitris Kalogeras, Dimitris Pantazatos, Mary Grammatikou, and Vasilis Maglaris. 2023. “SHAP Interpretations of Tree and Neural Network DNS Classifiers for Analyzing DGA Family Characteristics.” ResearchGate. January 2023. https://www.researchgate.net/publication/371590592_SHAP_Interpretations_of_Tree_and_Neural_Network_D
7. Liu, Mingxuan, Yilin Ning, Han Yuan, Marcus Eng, and Nan Liu. 2022. “Balanced Background and Explanation Data Are Needed in Explaining Deep Learning Models with SHAP: An Empirical Study on Clinical Decision Making.” *ArXiv.org*, June. https://doi.org/10.48550/arXiv.2206.04050.
8. Loyola-Gonzalez, Octavio. 2019. “Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses from a Practical Point of View.” *IEEE Access* 7: 154096–113. https://doi.org/10.1109/access.2019.2949286.
9. Lu, Shuyu, Ruoyu Chen, Wei Wei, Mia Belovsky, and Xinghua Lu. 2021. “Understanding Heart Failure Patients EHR Clinical Features via SHAP Interpretation of Tree-Based Machine Learning Model Predictions.” *AMIA … Annual Symposium Proceedings. AMIA Symposium* 2021: 813–22. https://pubmed.ncbi.nlm.nih.gov/35308970/.
10. Nochnack,Nick.2022. Image Classification. GitHub. https://github.com/nicknochnack/ImageClassification
11. Polanco, A. (2017). Medical MNIST Classification.Github. GitHub. https://github.com/apolanco3225/Medical-MNIST-Classification
12. Ren, Kui, Tianhang Zheng, Zhan Qin, and Xue Liu. 2020. “Adversarial Attacks and Defenses in Deep Learning.” *Engineering* 6 (3): 346–60. https://doi.org/10.1016/j.eng.2019.12.012.
13. Ribeiro, Marco, Sameer Singh, and Carlos Guestrin. 2016. “Model-Agnostic Interpretability of Machine Learning.” https://arxiv.org/pdf/1606.05386.
14. Sai, Siva, Uday Mittal, Vinay Chamola, Kaizhu Huang, Indro Spinelli, Simone Scardapane, Zhiyuan Tan, and Amir Hussain. 2023. “Machine Un-Learning: An Overview of Techniques, Applications, and Future Directions.” *Cognitive Computation* 16 (November): 482–506. https://doi.org/10.1007/s12559-023-10219-3.
15. Samek, Wojciech, Thomas Wiegand, and Klaus-Robert Müller. 2018. “Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models.” https://arxiv.org/pdf/1708.08296.
16. Shiri, Farhad Mortezapour, Thinagaran Perumal, Norwati Mustapha, and Raihani Mohamed. 2023. “A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU.”

17. Takemoto, Kazuhiro. 2024. “All in How You Ask for It: Simple Black-Box Method for Jailbreak Attacks.” *Applied Sciences* 14 (9): 3558. https://doi.org/10.3390/app14093558.
18. Traore, Boukaye Boubacar, Bernard Kamsu-Foguem, and Fana Tangara. 2018. “Deep Convolution Neural Network for Image Recognition.” Ecological Informatics 48 (November): 257–68. https://doi.org/10.1016/j.ecoinf.2018.10.002.
19. Tala Talaei Khoei, Hadjar Ould Slimane, and Naima Kaabouch. 2023. “Deep Learning: Systematic Review, Models, Challenges, and Research Directions.” *Neural Computing and Applications*, September. https://doi.org/10.1007/s00521-023-08957-4.
20. Van den Broeck, Guy, Anton Lykov, Maximilian Schleich, and Dan Suciu. 2022. “On the Tractability of SHAP Explanations.” *Journal of Artificial Intelligence Research* 74 (June): 851–86. https://doi.org/10.1613/jair.1.13283.
21. Yang, Wenli, Yuchen Wei, H Wei, Yanyu Chen, Guan Huang, Xiang Li, Renjie Li, et al. 2023. “Survey on Explainable AI: From Approaches, Limitations and Applications Aspects.” *Human-Centric Intelligent Systems* 3 (August): 161–88. https://doi.org/10.1007/s44230-023-00038-y.