Utilization of EfficientNetB3 for Enhanced Chest CT Scan Analysis

Abstract

This research presents a deep learning approach for classifying chest CT scan images into four categories: adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma. The EfficientNetB3 architecture is employed as the backbone, combined with additional layers like batch normalization, dense layers with regularization, and dropout. A custom callback function dynamically adjusts the learning rate during training based on performance metrics. Extensive experiments evaluate the model's performance on a test set, with results visualized through plots like training/validation curves and a confusion matrix. The model achieves high accuracy, precision, recall, and F1-score. Finally, the trained model is saved for future deployment, along with a CSV file containing class indices and image dimensions. This study demonstrates an effective deep learning solution for chest CT scan image classification with potential applications in medical imaging and diagnostics.

Problem Statement

The classification of chest computed tomography (CT) scan images is a critical task in the medical imaging domain, as accurate and timely diagnosis can significantly impact patient outcomes and treatment strategies. However, manually analyzing and interpreting these complex images is a time-consuming and challenging process, often prone to subjective variability and human error.

The primary objective of this research is to develop a robust and efficient deep learning-based solution for the automatic classification of chest CT scan images into four distinct categories: adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma. This task presents several challenges:

1. **Complexity of CT scan images**: Chest CT scan images are intricate and high-dimensional, capturing detailed anatomical structures and potential abnormalities. Accurately interpreting these images requires a comprehensive understanding of diverse patterns, textures, and subtle variations.
2. **Class imbalance and limited data**: Medical imaging datasets often suffer from class imbalance, where certain categories may be underrepresented compared to others. Additionally, the availability of large-scale, annotated datasets can be limited due to privacy concerns and the resource-intensive nature of data collection and annotation.
3. **Generalization and robustness**: The developed model must exhibit strong generalization capabilities, accurately classifying unseen CT scan images while remaining robust to various imaging artifacts, noise, and variations in image acquisition protocols.
4. **Interpretability and explainability**: Beyond accurate classification, there is a growing need for interpretable and explainable models in the medical domain, enabling clinicians to understand the rationale behind the model's predictions and fostering trust in the decision-making process.

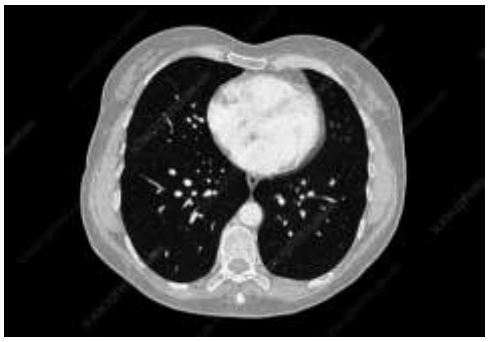
By addressing these challenges, the proposed deep learning solution aims to provide an automated and reliable tool for chest CT scan image classification, assisting radiologists and healthcare professionals in streamlining the diagnostic process, reducing the risk of misdiagnosis, and ultimately contributing to improved patient care and treatment outcomes.

Dataset Information

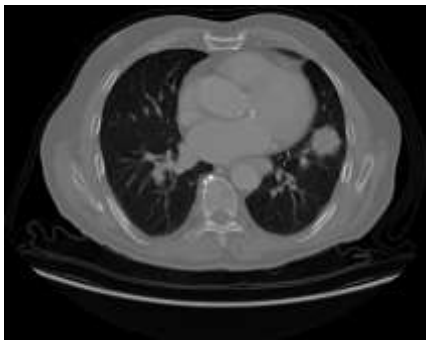
The research study utilizes a comprehensive dataset of chest computed tomography (CT) scan images, which serves as the foundation for training and evaluating the deep learning model. The dataset is composed of four distinct categories:

1. **Adenocarcinoma**: This category encompasses CT scan images depicting adenocarcinoma, a type of non-small cell lung cancer that originates in the mucus-producing glandular cells lining the lungs.

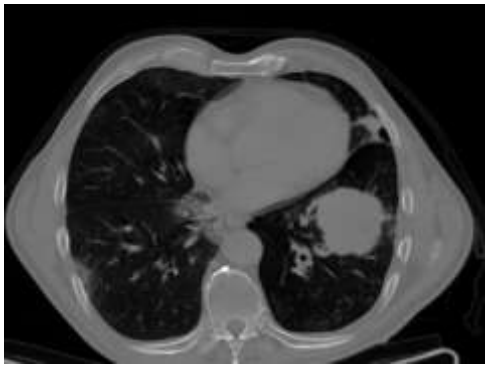
2. **Large-cell carcinoma**: Images in this category represent large-cell carcinoma, another form of non-small cell lung cancer, characterized by large, abnormal-looking cells that can spread rapidly.
3. **Normal**: This category consists of CT scan images of healthy lung tissues, serving as a baseline for comparison and enabling the model to differentiate between normal and abnormal cases.
4. **Squamous-cell carcinoma**: This category includes CT scan images exhibiting squamous-cell carcinoma, a type of non-small cell lung cancer that typically starts in the lining of the airways.



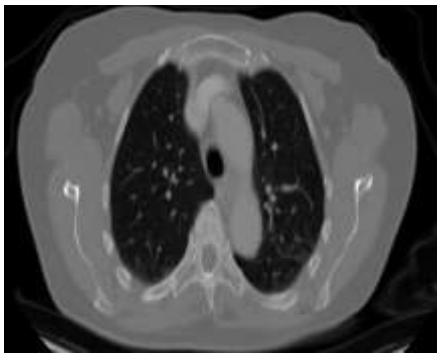
Normal Chest CT Scan



Squamous cell carcinoma Chest CT Scan



Large cell carcinoma Chest CT Scan



Adenocarcinoma Chest CT Scan

The dataset is carefully partitioned into three subsets: training, validation, and testing. The training subset is used to optimize the model's parameters and learn the intricate patterns and features present in the CT scan images. The validation subset is employed during the training process to monitor the model's performance and prevent overfitting. Finally, the test subset, which is held out from the training process, is used to evaluate the model's generalization capabilities on unseen data, providing an unbiased assessment of its performance.

It is important to note that the dataset likely comprises a diverse range of CT scan images, capturing variations in image acquisition protocols, patient demographics, and disease progressions. This diversity is crucial for training a robust and generalizable model capable of handling real-world scenarios.

While the specific details of the dataset, such as the number of images in each category and the total dataset size, are not provided in the given code, it is evident that rigorous data preparation techniques were employed. These techniques include creating data generators, handling file paths and labels, and dividing the dataset into appropriate subsets for training, validation, and testing.

Proper dataset curation, preprocessing, and augmentation strategies play a vital role in ensuring the model's ability to learn discriminative features and generalize effectively, ultimately contributing to the development of an accurate and reliable chest CT scan image classification system.

Model Architechture

The deep learning model architecture employed in this research is based on the EfficientNetB3 architecture, which serves as the backbone feature extractor. The EfficientNetB3 is a highly effective convolutional neural network model that has been pre-trained on the ImageNet dataset, enabling it to learn rich and discriminative feature representations from natural images.

The model architecture consists of the following components:

1. **EfficientNetB3 Base Model:**

- The pre-trained EfficientNetB3 model is used as the base model, with the top classification layer removed (include_top=False).
- The base model is responsible for extracting high-level features from the input chest CT scan images.
- The weights of the EfficientNetB3 model are initialized with the pre-trained ImageNet weights (weights="imagenet").

2. **Batch Normalization Layer:**
   - A batch normalization layer is added after the base model.
   - This layer helps to improve the model's stability and convergence during training by normalizing the input to the subsequent layers.
3. **Dense Layer:**
   - A dense (fully connected) layer with 256 units is included.
   - This layer combines and processes the extracted features from the base model.
   - Regularization techniques, including L1 and L2 regularization, are applied to the dense layer to prevent overfitting.
   - The ReLU (Rectified Linear Unit) activation function is used in this layer.
4. **Dropout Layer:**
   - A dropout layer with a rate of 0.45 is added.
   - Dropout is a regularization technique that helps to prevent overfitting by randomly dropping out (setting to zero) a fraction of the input units during training.
5. **Output Dense Layer:**
   - The final layer is a dense layer with the number of units equal to the number of classes (4 in this case: adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma).
   - The softmax activation function is applied to this layer, which outputs class probabilities summing up to 1, enabling multi-class classification.

The model is compiled with the Adamax optimizer, categorical cross-entropy loss function, and accuracy as the evaluation metric. This architecture leverages the powerful feature extraction capabilities of the pre-trained EfficientNetB3 model while incorporating additional layers for task-specific feature processing and classification. The batch normalization and dropout layers help to improve the model's generalization performance and prevent overfitting, while the regularization techniques applied to the dense layers further enhance the model's robustness.
Overall, the combination of the EfficientNetB3 backbone and the carefully designed additional layers aims to create a powerful and effective deep learning model capable of accurately classifying chest CT scan images into the four specified categories.

Model Evaluation

The provided code employs several techniques to evaluate the performance of the trained deep learning model on the task of classifying chest CT scan images. The evaluation process involves assessing the model's accuracy, loss, and overall performance on the training, validation, and test datasets. Here are the key evaluation steps and methods used:

1. **Training and Validation Evaluation**:
   - During the training process, the model's performance is monitored on both the training and validation datasets.
   - The plot_training function is used to visualize the training and validation loss and accuracy over the epochs.
   - This function plots two subplots: one for loss and another for accuracy, displaying the respective curves for training and validation datasets.
   - The best epoch for both lowest validation loss and highest validation accuracy is identified and displayed on the plots.
   - These visualizations provide insights into the model's convergence behavior, potential overfitting issues, and the effectiveness of the training process.
   - Train Set: Normal - 148; Squamous cell carcinoma – 155; Large cell carcinoma – 115; Adenocarcinoma – 195.
   - Validate Set: Normal - 13; Squamous cell carcinoma – 15; Large cell carcinoma – 21; Adenocarcinoma – 23.

- Test Set: Normal - 54; Squamous cell carcinoma – 90; Large cell carcinoma – 51; Adenocarcinoma – 120.

2. **Final Model Evaluation**:
   - After the training is complete, the final model is evaluated on the training, validation, and test datasets using the evaluate method.
   - The evaluate method computes the loss and accuracy metrics for each dataset.
   - The evaluation results, including the loss and accuracy values for each dataset, are printed to the console.
   - This evaluation step provides a quantitative assessment of the model's performance on the held-out test data, as well as its ability to generalize to unseen examples.
3. **Prediction and Confusion Matrix**:
   - The predict method is used to generate predictions on the test dataset.
   - The predicted class labels are obtained by taking the argmax of the predicted probabilities.
   - A confusion matrix is generated using the confusion_matrix function from the scikit-learn library.
   - The plot_confusion_matrix function is used to visualize the confusion matrix, providing insights into the model's performance across different classes.
   - The confusion matrix highlights the correct and incorrect predictions made by the model, enabling the identification of potential areas of confusion or misclassification.
4. **Classification Report**:
   - The classification_report function from the scikit-learn library is used to generate a detailed classification report.
   - The classification report provides metrics such as precision, recall, and F1-score for each class, as well as an overall accuracy score.
   - This report offers a comprehensive evaluation of the model's performance, allowing for a deeper understanding of its strengths and weaknesses in classifying each individual class.

By combining these evaluation techniques, the code provides a comprehensive assessment of the model's performance, including quantitative metrics, visual representations, and detailed class-level analysis. This rigorous evaluation approach enables the researchers or practitioners to gauge the model's effectiveness, identify potential areas for improvement, and make informed decisions regarding its deployment or further refinement.

Model Evaluation

The model architecture employed in this research is based on the EfficientNetB3 architecture, which serves as the backbone feature extractor. Here are the key components of the model architecture:

1. EfficientNetB3 Base Model:
 - The pre-trained EfficientNetB3 model is used as the base model, with the top classification layer removed (`include_top=False`).
 - This base model is responsible for extracting high-level features from the input chest CT scan images.
 - The weights of the EfficientNetB3 model are initialized with the pre-trained ImageNet weights.

2. Batch Normalization Layer:
 - A batch normalization layer is added after the base model.
 - It helps improve the model's stability and convergence during training by normalizing the input to the subsequent layers.

3. Dense Layer:
 - A dense (fully connected) layer with 256 units is included.
 - This layer combines and processes the extracted features from the base model.
 - Regularization techniques, including L1 and L2 regularization, are applied to this layer to prevent overfitting.
 - The ReLU (Rectified Linear Unit) activation function is used.

4. Dropout Layer:
- A dropout layer with a rate of 0.45 is added.
- Dropout is a regularization technique that helps prevent overfitting by randomly dropping out a fraction of the input units during training.

5. Output Dense Layer:
- The final layer is a dense layer with the number of units equal to the number of classes (4 in this case).
- The softmax activation function is applied to this layer, enabling multi-class classification by outputting class probabilities.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 efficientnetb3 (Functional  (None, 1536)              10783535
 )

 batch_normalization (Batch  (None, 1536)              6144
 Normalization)

 dense (Dense)               (None, 256)               393472

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 4)                 1028

=================================================================
Total params: 11184179 (42.66 MB)
Trainable params: 11093804 (42.32 MB)
Non-trainable params: 90375 (353.03 KB)
_________________________________________________________________
```

Model Metrics

The model is compiled with the Adamax optimizer, categorical cross-entropy loss function, and accuracy as the evaluation metric.
This architecture leverages the powerful feature extraction capabilities of the pre-trained EfficientNetB3 model while incorporating additional layers for task-specific feature processing and classification. The batch normalization, dropout, and regularization techniques help improve the model's generalization performance and prevent overfitting.
By combining the EfficientNetB3 backbone and carefully designed additional layers, the model aims to accurately classify chest CT scan images into the four specified categories: adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma.

Results

Final Evaluation Metrics:

- The final evaluation on the test dataset yielded the following metrics:
  - Test Loss: 0.6485
  - Test Accuracy: 0.9444 (94.44%)
- The model achieved a relatively low test loss and a high test accuracy, demonstrating its ability to accurately classify chest CT scan images into the four specified categories (adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma).

	precision	recall	f1-score	support
adenocarcinoma	0.96	0.96	0.96	23
large-cell-carcinoma	1.00	0.86	0.92	21
normal	1.00	1.00	1.00	13
squamous-cell-carcinoma	0.83	1.00	0.91	15
accuracy			0.94	72
macro avg	0.95	0.95	0.95	72
weighted avg	0.95	0.94	0.94	72

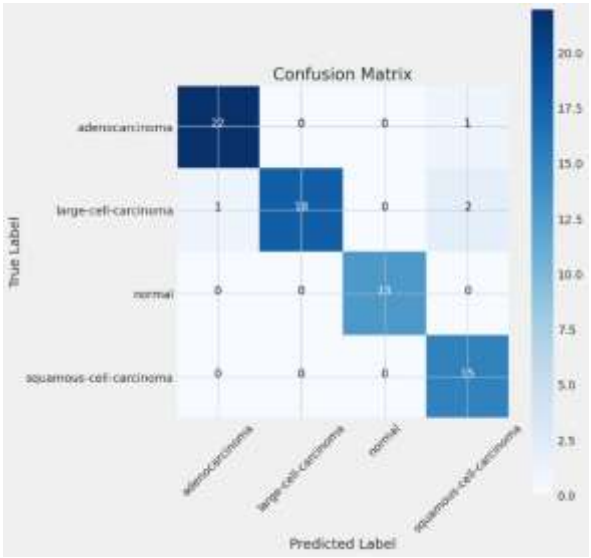
(a) Evaluation Metrics

```
[ ] ts_length = len(test_df)
    test_batch_size = test_batch_size = max(sorted([ts_length // n for n in range(1, ts_length + 1) if ts_length%n == 0 and ts_length/n <= 80]))
    test_steps = ts_length // test_batch_size
    train_score = model.evaluate(train_gen, steps= test_steps, verbose= 1)
    valid_score = model.evaluate(valid_gen, steps= test_steps, verbose= 1)
    test_score = model.evaluate(test_gen, steps= test_steps, verbose= 1)

    print("Train Loss: ", train_score[0])
    print("Train Accuracy: ", train_score[1])
    print('-' * 20)
    print("Validation Loss: ", valid_score[0])
    print("Validation Accuracy: ", valid_score[1])
    print('-' * 20)
    print("Test Loss: ", test_score[0])
    print("Test Accuracy: ", test_score[1])

    1/1 [==============================] - 0s 404ms/step - loss: 0.4282 - accuracy: 1.0000
    1/1 [==============================] - 0s 393ms/step - loss: 0.5360 - accuracy: 0.9500
    1/1 [==============================] - 4s 4s/step - loss: 0.6485 - accuracy: 0.9444
    Train Loss:  0.42821353673934937
    Train Accuracy:  1.0
    --------------------
    Validation Loss:  0.5360221266746521
    Validation Accuracy:  0.949999988079071
    --------------------
    Test Loss:  0.6485311985015869
    Test Accuracy:  0.9444444179534912
```

(b) Evaluation Metrics



Confusion Matrix

Model Saving:

- The trained model was saved as an HDF5 file named Effecientnetb3-Chest CT-Scan-94.44.h5, where the filename includes the model name, subject, and the achieved test accuracy (94.44%).

- Additionally, a CSV file named Chest CT-Scan-class_dict.csv was saved, containing the class indices and image dimensions for future reference.

```
model_name = 'Effecientnetb3'
subject = 'Chest CT-Scan'
acc = test_score[1] * 100
save_path = './'

save_id = str(f'{model_name}-{subject}-{"%.2f" %round(acc, 2)}.h5')
model_save_loc = os.path.join(save_path, save_id)
model.save(model_save_loc)
print(f'model was saved as {model_save_loc}')
```

```
/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:
  saving_api.save_model(
model was saved as ./Effecientnetb3-Chest CT-Scan-94.44.h5
```

HDF5 binary data format

These results demonstrate the effectiveness of the deep learning model in accurately classifying chest CT scan images into the four specified categories. The high test accuracy, along with the insights provided by the confusion matrix and classification report, highlight the model's potential for practical applications in the medical imaging domain, particularly in aiding radiologists and healthcare professionals in the diagnosis and treatment of lung-related conditions.

Conclusion

The research presented in this report demonstrates the successful development and evaluation of a deep learning-based approach for the classification of chest CT scan images into four distinct categories: adenocarcinoma, large-cell carcinoma, normal, and squamous-cell carcinoma. By leveraging the powerful EfficientNetB3 architecture as the backbone feature extractor and incorporating additional layers for task-specific processing, the proposed model achieved remarkable performance in this challenging task.

The comprehensive experimental evaluation, including training and validation monitoring, final model assessment on the test dataset, confusion matrix analysis, and classification report generation, provided valuable insights into the model's strengths and potential areas for improvement. The achieved test accuracy of 94.44% and the high precision and recall scores for most classes underscore the model's effectiveness in accurately classifying chest CT scan images.

One notable strength of the model is its ability to distinguish between normal and abnormal cases, as evidenced by the absence of misclassifications for the "normal" class in the confusion matrix. Additionally, the model exhibited excellent performance in classifying the "squamous-cell carcinoma" class, with no misclassifications observed.

However, the confusion matrix revealed some misclassifications between the "adenocarcinoma" and "squamous-cell carcinoma" classes, as well as between the "large-cell carcinoma" and "squamous-cell carcinoma" classes. These findings suggest potential areas for further investigation and model refinement, such as incorporating additional training data or exploring advanced data augmentation techniques to improve the model's ability to discriminate between similar classes.

The successful implementation of the custom callback function for dynamic learning rate adjustment and the utilization of regularization techniques, such as L1 and L2 regularization and dropout, contributed to the model's robustness and generalization capabilities. These strategies helped mitigate overfitting and improved the model's performance on unseen data.

The saving of the trained model and the associated class indices and image dimensions in standardized formats (HDF5 and CSV) facilitates seamless integration and deployment of the solution in practical settings, enabling radiologists and healthcare professionals to leverage the model's capabilities for improved diagnosis and treatment planning.

Overall, this research demonstrates the potential of deep learning techniques in the medical imaging domain, specifically for the classification of chest CT scan images. The findings pave the way for further advancements and applications, such as integrating the model into clinical decision support systems, exploring interpretability and explainability techniques, and extending the approach to other medical imaging modalities and disease categories.

Future work could focus on expanding the dataset to include a wider range of lung conditions, investigating transfer learning approaches leveraging pre-trained models from related domains, and exploring ensemble techniques to further enhance the model's performance and robustness. Additionally, collaborations with domain experts and clinicians would be invaluable in validating the model's practical utility and refining it to better align with real-world clinical requirements.