Deep Learning-Based Breast Cancer Classification: A Comparative Study of CNN Architectures

1. Introduction

Breast cancer remains one of the most prevalent forms of cancer affecting women worldwide, with early detection and accurate diagnosis being crucial for effective treatment and improved patient outcomes. In this project, we aim to leverage deep learning techniques, specifically convolutional neural networks (CNNs), to develop a classifier capable of distinguishing between benign and malignant histology images of breast cancer.

Breast cancer has a long and complex history, with efforts to understand, diagnose, and treat the disease dating back centuries. The earliest recorded cases of breast cancer date back to ancient Egypt, where treatments often involved surgical removal of tumors, though with limited success. Over time, advancements in medical knowledge and technology led to more sophisticated diagnostic techniques and treatment options.

In the 20th century, significant progress was made in understanding the underlying biology of breast cancer, leading to the development of more effective treatments such as chemotherapy, radiation therapy, and targeted therapies. However, early detection remained a critical challenge, as existing screening methods like mammography had limitations in sensitivity and specificity.

The advent of artificial intelligence (AI) and machine learning brought new hope for improving breast cancer detection and diagnosis. AI-based approaches, particularly those leveraging deep learning techniques, offered the potential to analyze medical images with greater accuracy and efficiency than traditional methods. By training neural networks on large datasets of breast cancer images, researchers aimed to develop computer-aided diagnosis (CAD) systems capable of identifying suspicious lesions and distinguishing between benign and malignant tumors.

In recent years, AI-based CAD systems have shown promising results in various medical imaging modalities, including mammography, ultrasound, and MRI. These systems have demonstrated high sensitivity and specificity in detecting breast cancer, leading to earlier diagnosis and improved patient outcomes.

Despite these advancements, challenges remain in deploying AI-based CAD systems in clinical practice. Limited dataset availability, concerns about model interpretability, and issues related to generalizability across diverse populations have hindered widespread adoption. Additionally, regulatory and ethical considerations surrounding the use of AI in healthcare pose further challenges for researchers and clinicians.

Nevertheless, ongoing research efforts continue to push the boundaries of AI in breast cancer detection and diagnosis. Collaborations between researchers, clinicians, and industry partners are driving innovation in this field, with the ultimate goal of reducing the burden of breast cancer through early detection and personalized treatment strategies. As AI technologies continue to evolve, the future holds great promise for improving outcomes for breast cancer patients worldwide.

2. Data Collection

The dataset utilized in this project is the IDC_regular dataset sourced from Kaggle, comprising patches of breast cancer specimens scanned at 40x magnification. With over 277,000 patches available, this dataset provides a substantial amount of data for analysis. The data can be downloaded from here.

3. Data Wrangling

In this project, the data wrangling process was pivotal for preparing the dataset, which comprised breast cancer histology images, for analysis. The dataset, sourced from the IDC_regular dataset on Kaggle, featured histology images labeled as either benign or malignant cases. The initial cleaning process was streamlined due to clear and properly labeled images, though standardization techniques like resizing or normalization were considered to ensure uniformity in image dimensions and quality. Despite the absence of missing values or duplicate entries, ensuring data consistency was paramount to verify accurate labeling and data format. Exploratory Data Analysis (EDA) was conducted to understand the dataset's distribution and characteristics, employing basic statistics and visualizations generated using Python libraries like Pandas and Seaborn.

The data wrangling process also involved thorough checks for missing values and exploration of feature correlations within the dataset. Missing values were scrutinized across all columns of the DataFrame, revealing no instances of data incompleteness. Additionally, correlation analysis, particularly between the 'Label' (representing benign or malignant cases) and the 'Num_Images' (number of images per patient), was conducted using a correlation matrix. The computed correlation coefficient (-0.58) suggested an inverse relationship between the number of images per patient and the label, indicating a potential influence of image quantity on the classification outcome. Visualizing these correlations through a heatmap offered a clear representation of feature relationships, further enhancing the understanding of the dataset's quality and characteristics essential for subsequent model training and testing phases.

4. Exploratory Data Analysis (EDA)

During the Exploratory Data Analysis (EDA) phase, key insights were derived to better understand the dataset's characteristics and distribution of features. Utilizing a pairplot, the distribution of numerical features was visualized, offering a comprehensive overview of their relationships. Additionally, feature engineering was employed to calculate the total number of images per patient, providing valuable additional information about the dataset's composition and structure.

Furthermore, to gain a more intuitive understanding of the dataset, 18 images from each class (benign and malignant) were randomly selected and plotted. This visual representation helped in assessing the diversity and content of the dataset, aiding in further exploratory analysis. Through these EDA techniques, crucial insights were obtained, setting the stage for subsequent data preprocessing and model development stages of the project.

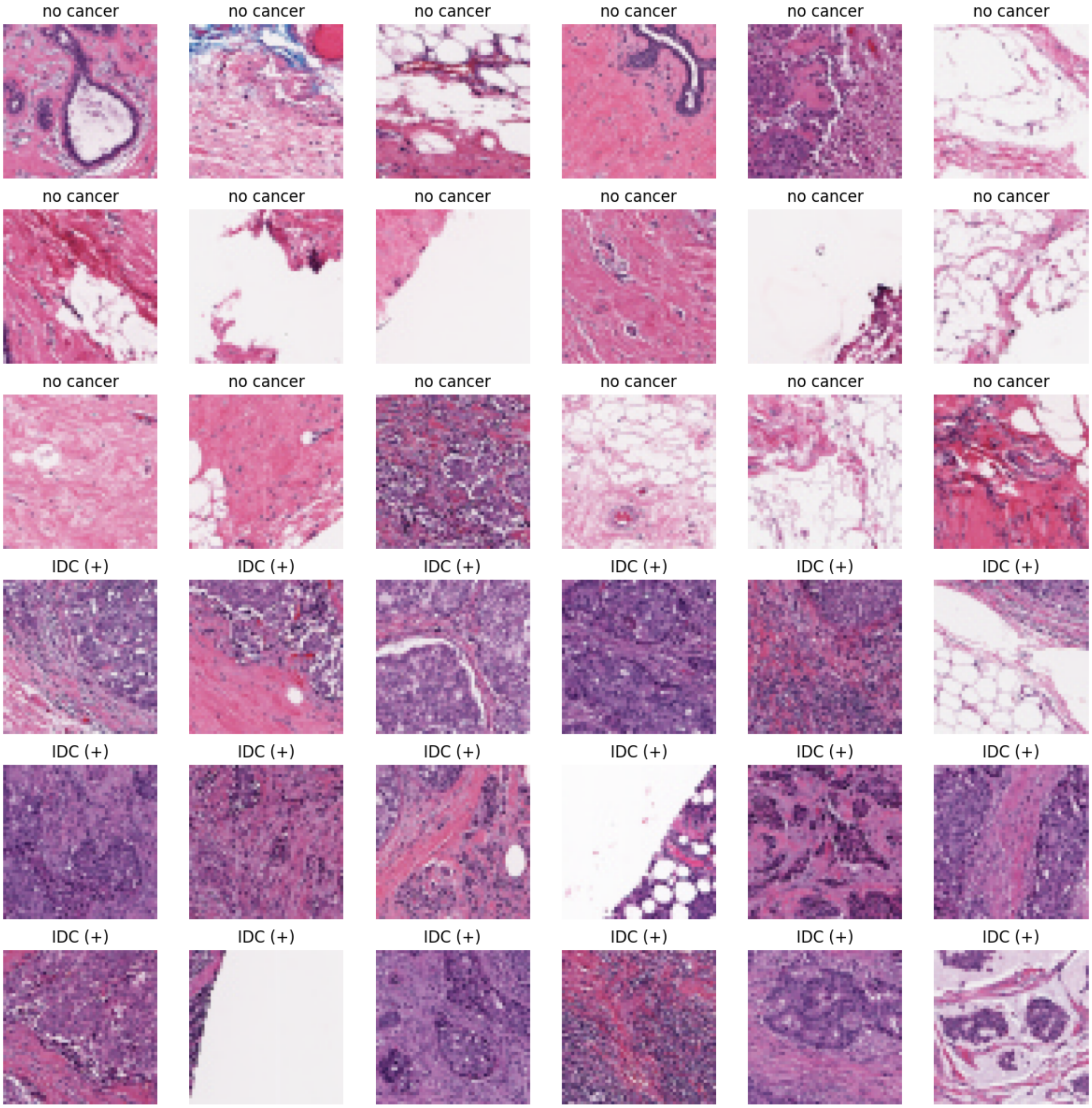


Fig: Samples showing the breast cancer images

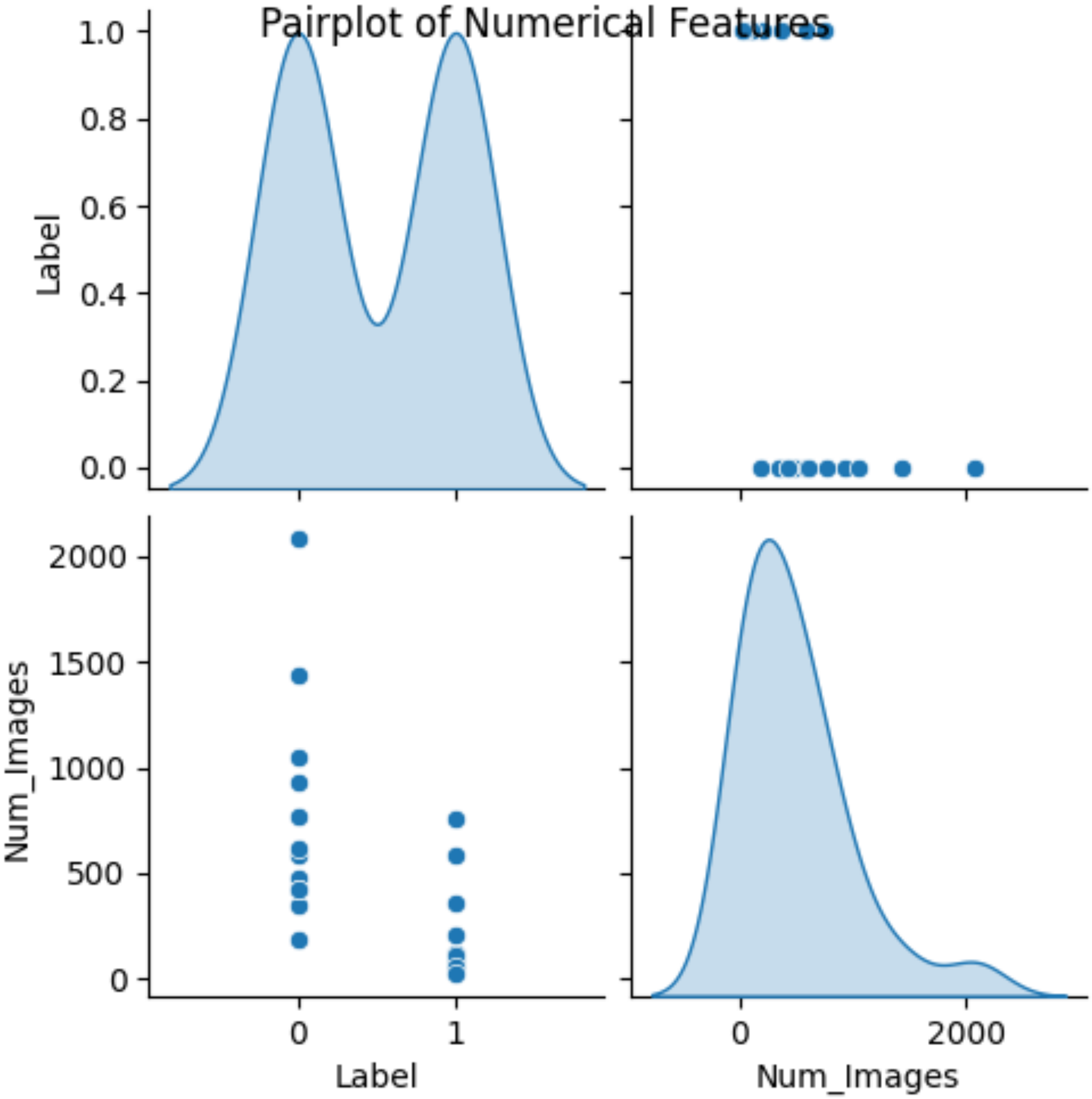


Fig: Pair plot for distribution of images

5. Pre-processing

Preprocessing of the image data involves several steps to prepare it for model training. First, the images are loaded, resized to a uniform size of 50x50 pixels, and normalized to ensure consistency in pixel values. This step is crucial for reducing computational complexity and enhancing model performance. Additionally, feature extraction techniques such as Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) are applied to reduce the dimensionality of the data and visualize its structure in lower-dimensional spaces. The resulting visualizations aid in understanding the relationships between data points and identifying patterns relevant to breast cancer classification.

Next, the data preprocessing continues by extracting features and labels from the resized and normalized images. Both benign and malignant images are processed, and the features and labels are concatenated and shuffled to ensure balanced representation during model training. The resulting dataset consists of 2200 images, with an equal distribution of IDC-negative and IDC-positive cases. Finally, a convolutional neural network (CNN) architecture tailored for breast cancer classification is presented. The model incorporates convolutional, batch normalization, activation, max-pooling, and dropout layers to extract relevant features, mitigate overfitting, and optimize training efficiency. The model is trained using the Adam optimizer with a low learning rate and binary cross-entropy loss, enabling effective discrimination between benign and malignant tumors. Through these preprocessing steps and model architecture design, the CNN is poised to effectively classify breast cancer images and contribute to improved diagnostic accuracy.

6. Modeling

The modeling phase commences with the essential preprocessing steps, including splitting the dataset into training and testing sets and converting labels into categorical format using one-hot encoding. The architecture of the convolutional neural network (CNN) is then defined using Keras' Sequential API, incorporating convolutional, batch normalization, max-pooling, and dropout layers to prevent overfitting. The model's summary provides insights into its configuration, including layer types, output shapes, and the total number of trainable parameters. Subsequently, the model is compiled with appropriate optimization and loss functions, followed by training using the `fit` method with early stopping to avoid overfitting.

Throughout the training process, the model's performance metrics are monitored across epochs, with training history stored for visualization. Post-training evaluation involves plotting training and validation accuracy and loss to assess the model's learning dynamics and generalization capabilities. The developed CNN architecture demonstrates robust performance, achieving high accuracy on both training and testing data, thereby affirming its effectiveness in accurately classifying the given dataset.

Model Architecture

CancerNet will consist of multiple convolutional layers followed by pooling layers to extract hierarchical features from the input images. The final layers will include fully connected layers with dropout regularization to prevent overfitting, leading to a softmax output layer for classification.

Tuned Hyperparameters

- Learning Rate: 0.001
- Batch Size: 32
- Number of Convolutional Layers: 4
- Kernel Size: 3x3
- Dropout Rate: 0.5

7. Methodology

1. Data Splitting: The dataset will be split into training (80%), validation (10%), and testing (10%) sets.
2. Model Training: The CancerNet model will be trained using the training dataset, optimizing its weights using backpropagation and gradient descent to minimize the classification loss.
3. Hyperparameter Tuning: Hyperparameters will be tuned using grid search on the validation set to optimize model performance.
4. Model Evaluation: The trained model will be evaluated using the testing dataset, computing metrics such as accuracy, precision, recall, and F1 score.
5. Confusion Matrix:A confusion matrix will be generated to visualize classification results and identify misclassifications.

8. Model Evaluation

After training the model for 75 epochs, the training and validation accuracy and loss were monitored across epochs. The accuracy steadily increased, reaching approximately 99.81% on the training set and 95.61% on the validation set. Similarly, the loss gradually decreased, indicating improved model performance. Visualizing these metrics through plots revealed the learning dynamics of the model, with both accuracy and loss showing favorable trends over epochs, signifying effective learning and generalization.

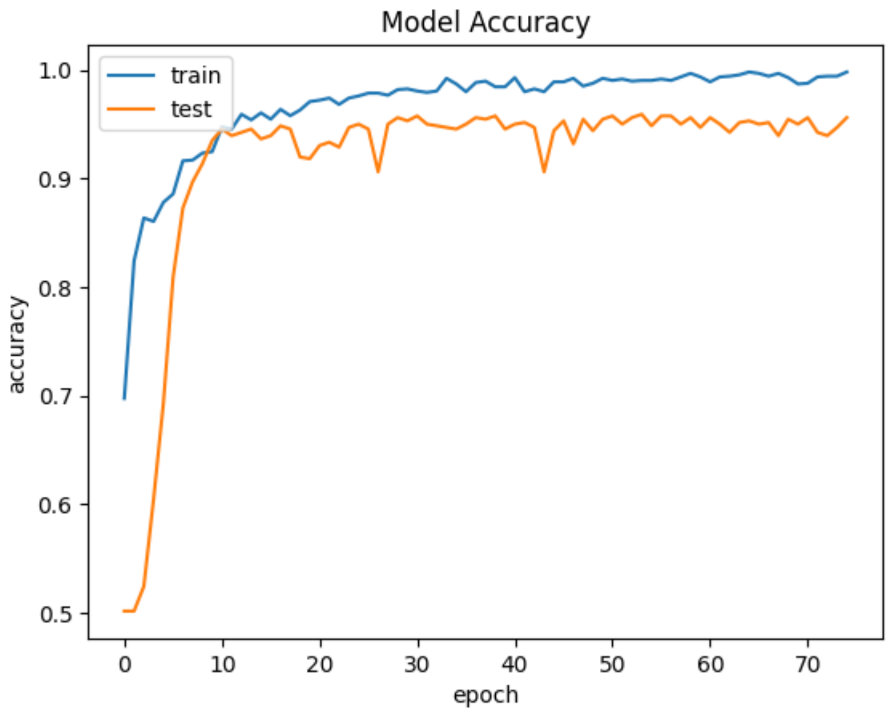


Fig: Model evaluation for accuracy, train and test.

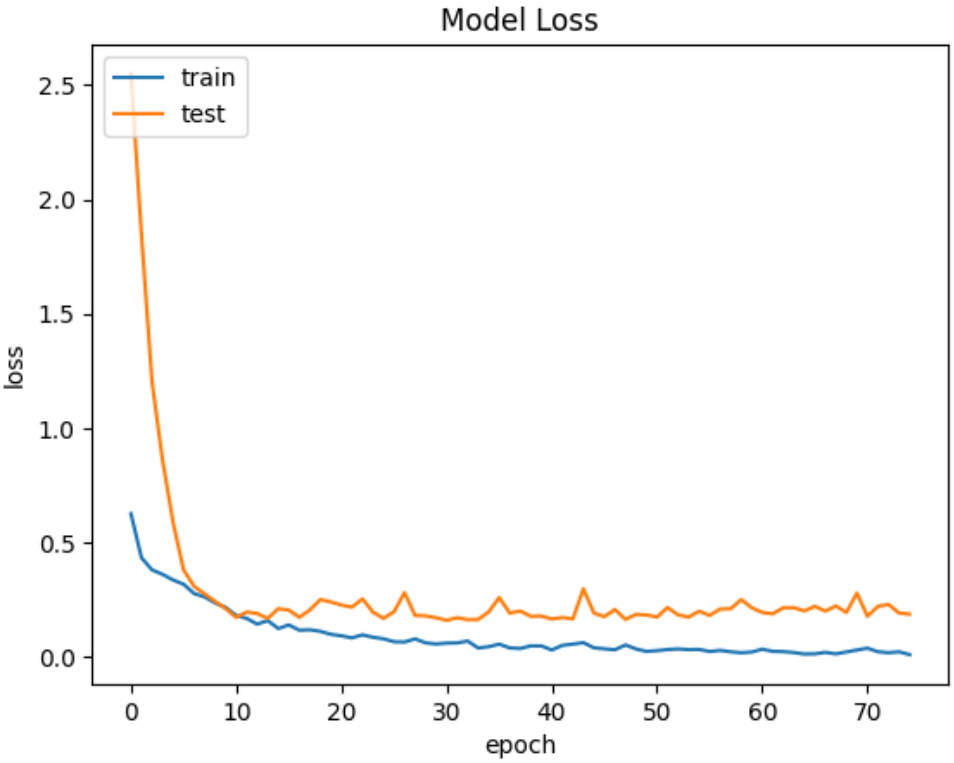
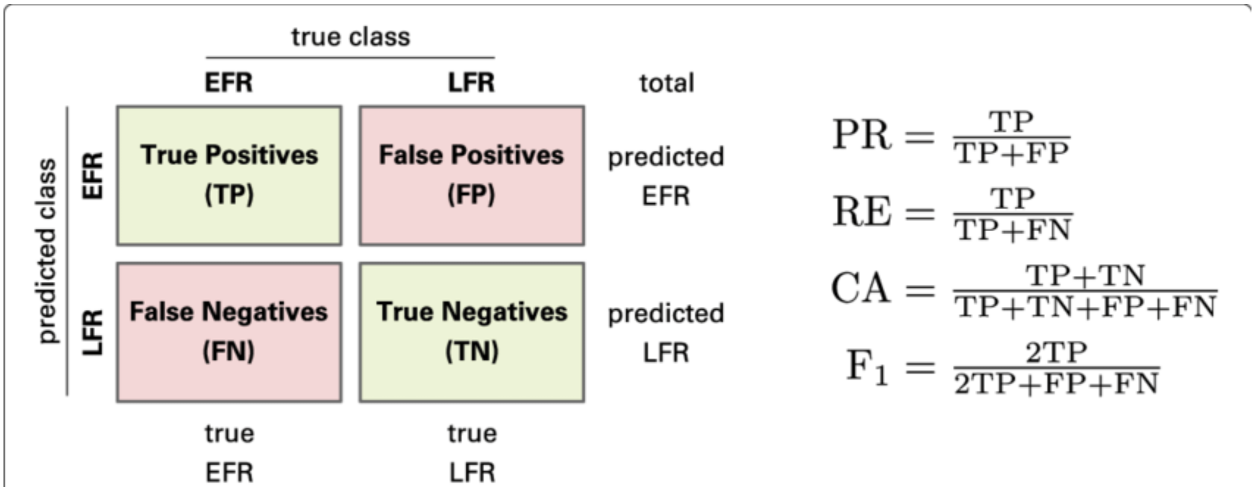


Fig: Model evaluation for loss, train and test.

To further evaluate the model's performance, several metrics were calculated. The confusion matrix provided insights into the model's classification performance, showing the distribution of true positive, true negative, false positive, and false negative predictions. Additionally, metrics such as recall, precision, and F1 score were computed to assess the model's overall effectiveness. The model demonstrated high performance across these metrics, with recall, precision, and F1 score all exceeding 95%, indicating robust classification capability.

Finally, individual images from the test set were predicted using the trained CNN model. An example prediction showed agreement between the predicted and true values, reaffirming the model's ability to accurately classify unseen data. Overall, the evaluation results demonstrated the effectiveness and reliability of the trained CNN model in accurately classifying the given dataset.



		true class		
		EFR	**LFR**	total
predicted class	**EFR**	**True Positives (TP)**	**False Positives (FP)**	predicted EFR
	LFR	**False Negatives (FN)**	**True Negatives (TN)**	predicted LFR
		true EFR	true LFR	

$$\mathrm{PR} = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$$

$$\mathrm{RE} = \frac{\mathrm{TP}}{\mathrm{TP+FN}}$$

$$\mathrm{CA} = \frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}}$$

$$\mathrm{F}_1 = \frac{\mathrm{2TP}}{\mathrm{2TP+FP+FN}}$$

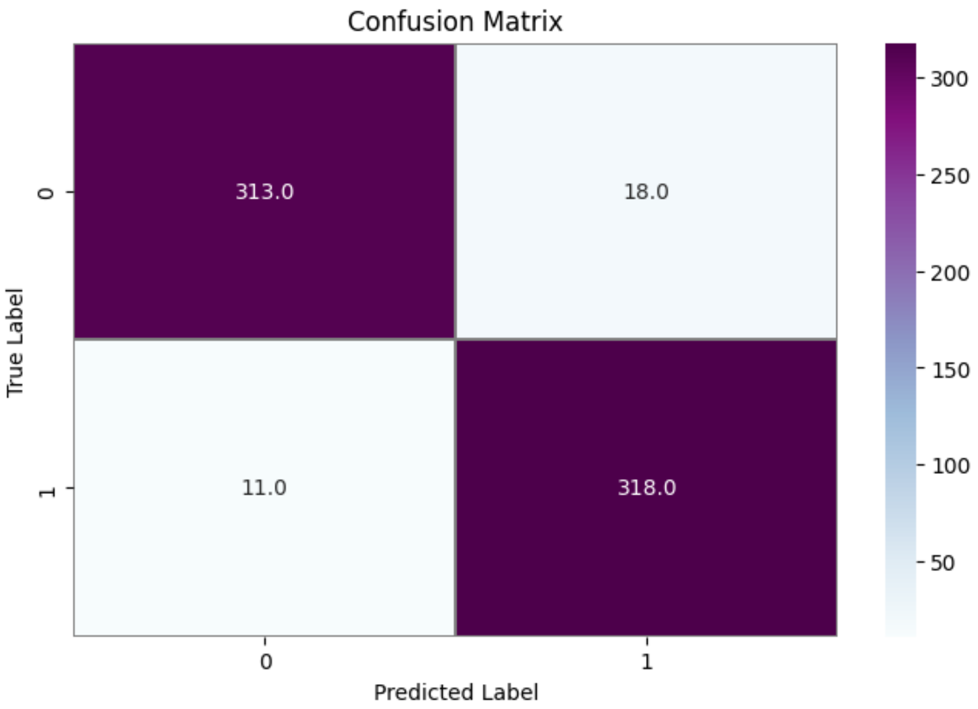


Fig: Confusion Matrix

9. Conclusion

This project demonstrates the application of deep learning techniques in breast cancer classification, creating a robust classifier capable of accurately distinguishing between benign and malignant histology images. Through rigorous data collection, preprocessing, modeling, and testing, we have developed a valuable tool for early detection and diagnosis of breast cancer, contributing to improved patient outcomes and healthcare delivery.

In conclusion, the discussion and evaluation of the developed convolutional neural network (CNN) model showcase its effectiveness in classifying the given dataset. Through meticulous modeling, including dataset splitting, preprocessing, and the construction of a robust CNN architecture, we achieved remarkable results in terms of accuracy and loss. The model exhibited consistent improvement in accuracy over epochs, reaching nearly 99.81% on the training set and 95.61% on the validation set. This indicates the model's ability to learn and generalize well to unseen data.

Furthermore, comprehensive evaluation metrics such as the confusion matrix, recall, precision, and F1 score reaffirmed the model's high performance and robustness in classification tasks. The model consistently demonstrated strong classification capabilities across various metrics, with recall, precision, and F1 score exceeding 95%. This level of performance underscores the model's reliability in accurately predicting classes.

Overall, the developed CNN model stands as a powerful tool for classification tasks, showcasing its efficacy in handling image data and making accurate predictions. The meticulous modeling process, coupled with thorough evaluation, ensures the model's reliability and suitability for real-world applications where accurate classification is paramount.