Topic: Bring It All Together Alice Tang and Yinda Chen

TF Lite Breast Cancer Detection Week 13: Bring It All Together

Stage I: Define Business Objective

1. Problem Statement

Breast cancer is the most common cancer in women andd second leading cause of cancer-related deaths globally. Early and accurate detection significantly improves survival rates, yet current screening methods face critical limitations.

1. **Missed Diagnoses:** Up to **50% of breast cancers go undetected** in patients with dense breast tissue.
2. **False Negatives and Positives:** Approximately **25% of cancers are misdiagnosed within two years of screening**, leading to delayed interventions or unnecessary treatments.
3. **Time-Intensive Analysis:** Manual mammogram interpretation can take **up to 30 minutes per scan**, creating inefficiencies and backlogs in diagnostic workflows.

2. Business Objective

To address these issues, the business objective is to develop a solution that:

1. **Enhances diagnostic accuracy** by reducing false positives and false negatives.
2. **Improves operational efficiency** by minimizing the time needed to analyse mammograms.
3. **Delivers measurable economic value** through cost savings for patients, healthcare providers, and society.
    a. **Per-Patient Cost Savings:** By reducing false negatives and false positives, the solution achieves an estimated **$31,737.39 per patient in savings**:
        i. **False Positives**: $12,880.30 saved per patient.
        ii. **False Negatives**: $18,850 saved per patient.
        iii. **Radiologist Time Savings**: $7.09 saved per mammogram (25 minutes saved per scan).
    b. **Societal Cost Reductions:** Annual reductions in societal costs are estimated at **$186.02 million**:
        i. **False Positive Losses**: $92.27 million reduced annually.
        ii. **False Negative Losses**: $93.75 million reduced annually.

4. **Promotes better patient outcomes** by facilitating earlier detection and treatment, leading to **higher survival rates** and the **reduced need for invasive and costly treatments**.

Stage 2: Define Model Objective

To achieve these business goals, the model objective is to build a machine learning-based TensorFlow Lite application to assist radiologists in mammogram analysis. The goal is to:

1. **Identify abnormalities** in mammograms with high precision, including subtle patterns that radiologists may miss.
2. **Reduce diagnostic errors** (false positives and false negatives) to improve reliability.
3. **Accelerate diagnostic workflows** by cutting analysis time from **30 minutes** to approximately **5 minutes**.
4. **Support radiologists** as an assistive tool, particularly for patients with dense breast tissue where detection is more difficult.

Stage 3: Data Assessment and Preprocessing

The goal of this stage was to properly prepare the datasets for model training by ensuring data quality, consistency, and readiness, inccluding handling missing values, encoding categorical variables, cleaning metadata, and preprocessing image data.

3.1 Preprocessing for the dicom_data Dataset

The dicom_data dataset contains metadata related to mammogram images. The preprocessing steps for this dataset focused on cleaning the metadata and ensuring the image paths were correctly aligned.

- **Data Cleaning**: Removed irrelevant columns (e.g., PatientBirthDate, StudyDate) to focus on key metadata (e.g., SeriesDescription, Laterality).
- **Handling Missing Data**: Applied forward and backward fill for missing values in key columns.
- **Image Paths**: Corrected image paths based on the directory structure and synchronized metadata.

3.2 Preprocessing for the calc_case Dataset

The calc_case dataset contains cases related to calcifications in mammograms. Preprocessing focused on cleaning the data, encoding categorical variables, and addressing missing values.

- **Data Cleaning**: Created a deep copy and converted columns to appropriate data types (e.g., categorical columns to 'category').

- **Missing Data**: Applied backward fill for columns like pathology and calc_type.
- **One-Hot Encoding**: Converted categorical variables (e.g., calc_type) into binary format.

3.3 Preprocessing for the mass_case Dataset

The mass_case dataset contains cases of masses in mammograms. Similar to calc_case, preprocessing focused on cleaning, handling missing data, and encoding categorical variables.

- **Data Cleaning**: Created a working copy and converted columns to categorical types.
- **Missing Data**: Applied backward fill for mass-related columns.
- **One-Hot Encoding**: Applied to categorical features like mass_shape and mass_margin.

3.4 Preprocessing for Image Data

Images play a key role in the project for detecting breast cancer. The images were preprocessed based on their categories (full mammogram, cropped, and ROI mask images) to ensure consistent format and quality for the models.

- **Image Categories:** Worked with full mammogram, cropped, and ROI mask images.
- **Image Resizing:** Resized images to 256x384 (mammograms), 256x256 (cropped), and 2400x3600 (ROI masks).
- **Preprocessing Techniques:** Applied grayscale conversion, contrast enhancement, and noise reduction for mammograms, and Gaussian blurring and histogram equalization for cropped images.

3.5 Missing Data & Outliers

Missing numerical data was imputed to avoid loss, while categorical data was encoded. Outliers were detected via Z-scores and handled based on domain knowledge.

3.6 Feature Transformation & Engineering

Created new features (e.g., merging calc_type and calc_distribution) and applied normalization/scaling to numerical features for model optimization.

3.7 Dimensionality Reduction

Used Fisher Score to identify significant features which removed irrelevant ones to improve model performance.

3.8 Key Challenges and Takeaways

- **Missing Data:** Effectively imputed with forward/backward filling.
- **Memory Optimization:** Reduced memory usage by converting columns and resizing images.
- **Categorical Encoding:** Applied One-Hot Encoding to ensure proper representation of categorical data.

- **Image Preprocessing:** Tailored preprocessing for different image types to ensure quality and consistency.

Stage 4: Model Development

4.1 Model Selection and Rationale

For this project, we selected the EfficientNetV2B0 architecture for breast cancer detection in mammography images. This model was chosen for its balance between accuracy and computational efficiency, an essential feature given the real-time nature of the intended TensorFlow Lite application. EfficientNetV2B0 is particularly well-suited for mobile and edge devices due to its compact architecture, while still maintaining strong performance on complex tasks like medical image classification. The EfficientNetV2B0 model includes compound scaling, a method that adjusts the depth, width, and resolution of the network in a way that optimizes performance with minimal computational burden. This allowed us to effectively use the pre-trained ImageNet weights through transfer learning, enabling faster convergence and more accurate results on our specific mammography dataset.

Initially, we also explored a CNN model for classifying breast cancer images. We were interested in using this model due to the nature of the data and the strengths of CNNs in image classification tasks. However, we did not yield strong results. We then switched to transfer learning, using a pre-trained ResNet50 model to detect breast cancer in mammography images. ResNet50, a deep CNN pre-trained on the ImageNet dataset, captures a wide variety of features. By freezing some of its layers and adding custom fully connected layers, we fine-tuned the model for classifying mammograms as benign or malignant. We believe the model performed significantly better than the CNN due to the integration of data augmentation techniques, such as random flipping, brightness adjustment, and contrast and saturation manipulation. These techniques improved generalization and reduced overfitting— especially with our imbalanced dataset. Additionally, we drew inspiration from Kaggle user "hayder17"'s notebook for preprocessing, which had a notable positive impact on our model's performance, further emphasizing the importance of proper data preprocessing.

Ultimately, we decided to proceed with the EfficientNetV2B0 model due to its superior performance and computational efficiency, which aligns better with the project's real-time deployment needs.

.

4.2 Model Training and Hyperparameter Tuning

Training the EfficientNetV2B0 model required careful selection of hyperparameters to optimize both the performance and computational efficiency of the network.

4.2.1 Hyperparameter Selection

Several key hyperparameters were tuned throughout the model development process, including:

- **Dropout Rate:** We tested different dropout rates (0.1, 0.2, 0.3) to mitigate overfitting. Dropout is a regularization technique that randomly disables a fraction of neurons during training, forcing the model to learn more general features.
- **Trainable Layers:** To strike a balance between leveraging pre-trained features from ImageNet and adapting to the specifics of our dataset, we experimented with different numbers of trainable layers (9, 20, 30). A higher number of trainable layers allows the model to better adapt to the target dataset, but is also at risk of overfitting if not handled properly.
- **Learning Rate:** We implemented an adaptive learning rate strategy using the ReduceLROnPlateau callback, which reduces the learning rate once validation loss plateaus. This helps the model avoid getting stuck in local minima and allows for stable convergence.

4.2.2 Training Strategy

The model was trained using the following approach:

- **Transfer Learning:** We initialized the EfficientNetV2B0 model with pre-trained weights from ImageNet and then fine-tuned the model on the mammography dataset. The weights from the initial layers were frozen to retain the general features learned from ImageNet, while the deeper layers were unfrozen and trained to specialize on mammography data.
- **Epochs and Batch Size:** The model was trained for 25 epochs with a batch size of 32, chosen based on the available computational resources and dataset size. The training process was computationally intensive but ensured that the model had enough iterations to converge effectively.

4.3 Model Selection and Final Configuration

After testing various combinations of dropout rates and trainable layer configurations, we selected the optimal model configuration based on the highest validation accuracy. The best-performing configuration was achieved with:

- **Dropout Rate:** 0.1
- **Trainable Layers:** 20
- **Validation Accuracy:** 0.9838

This configuration exhibited a good balance between generalization and performance, with minimal overfitting observed during training.

4.4 Comparison of Each Model's Metrics

The following table presents the validation loss and accuracy for each of the nine models:

Model	Validation Loss	Validation Accuracy
CNN Model 1	1.624226	0.431972802
CNN Model 2	1.950322	0.564626
CNN Model 3	1.535782	0.564626
ResNet Model 1	0.159271	0.956731
ResNet Model 2	0.140743	0.950321
ResNet Model 3	0.215971	0.947115
EfficientNet Model 0.3 Dropout Rate with 20 Layers	0.0698	0.9808
EfficientNet Model 0.1 Dropout Rate with 30 Layers	0.0643	0.9811
EfficientNet Model 0.1 Dropout Rate with 20 Layers	0.0610	0.9838

4.4 Performance Analysis

In our final evaluation, the model with the chosen hyperparameters demonstrated strong performance with a **validation accuracy of 98.38%**. The model showed consistent improvement over the 25 training epochs, with validation loss steadily decreasing. These results indicate that the model is effective in detecting breast cancer from mammography images, capable of distinguishing malignant from benign cases with high accuracy.

Stage 5: Model Validation

After training the EfficientNetV2B0 model with the selected hyperparameters, we proceeded with validating its performance on a separate validation set to assess its generalization capabilities.

5.1 Performance Metrics

To ensure the model's effectiveness in detecting breast cancer from mammography images, we evaluated its performance using several key metrics:

- **Accuracy**: The proportion of correct predictions (both benign and malignant) out of all predictions.
- **Precision**: The proportion of true positives (correct malignant classifications) among all predicted positives, important for avoiding false positives.
- **Recall**: The proportion of true positives over all actual positive cases, ensuring the detection of as many malignant cases as possible.
- **F1-Score**: The harmonic mean of precision and recall, providing a balanced measure especially in the case of our imbalanced dataset.
- **AUC-ROC Curve**: The area under the curve indicating the trade-off between sensitivity (true positive rate) and specificity (false positive rate). A higher AUC suggests better model performance in distinguishing malignant from benign cases.

5.2 Validation Performance

To validate the model's performance, we used a dedicated validation set separate from the training data. This allowed us to assess how well the model generalized to new data. We monitored both training and validation accuracy/loss during the 25 epochs of training, making sure that the model did not overfit to training data. The validation accuracy reached 98.38%, which implies that the model successfully generalized and was able to effectively distinguish between benign and malignant cases.

5.3 Error Analysis and Class-wise Performance

Given the class imbalance in the dataset, we conducted a detailed error analysis to better understand the model’s performance on different classes:

- **Confusion Matrix:** A confusion matrix was generated to visualize the number of true positives, false positives, true negatives, and false negatives. This helped identify which specific cases the model struggled to make accurate predictions on.
- **Class-wise Performance:** We also examined the model’s precision, recall, and F1-score for each class, ensuring that it performed well across all categories (benign and malignant masses, calcifications), we paid special attention to the malignant class, which is of the utmost importance for sensitive cases like medical imagery.
- **Data Shuffling:** From our error analysis, we also identified a key issue in the training process related to data shuffling. We set the **shuffle = False** for the training validation, and test datasets.

This decision negatively affected model performance and its ability to generalize. By not shuffling the data, the model may have been exposed to the same sequences of images in each epoch— thereby resulting in overfitting and poor generalization to new data. After correcting this by setting **shuffle = True** during training and **shuffle = False** for validation and testing, we saw significant improvement in model performance, including better generalization, reduced overfitting, and more reliable evaluation metrics.

Stage 6: Deficiency Assessment and Limitations

In the process of developing the breast cancer detection model, we encountered multiple challenges that impacted the overall performance and reliability of the system. Despite implementing multiple techniques to address the data imbalance and enhance the model's accuracy, we still faced certain limitations. These shortcomings, that range from data handling issues to the complexity of the model architecture, highlight any areas for further improvement and refinement in future iterations of the project.

The following outlines the key deficiencies and limitations encountered throughout the development process:

1. **Class Imbalance**: Despite efforts to balance the dataset, the high proportion of non-sepsis cases likely reduced the model's sensitivity and precision in detecting the minority class.
2. **Ethical Biases and Concerns:** Features like breast density, which correlate with age and other protected attributes, raise ethical concerns. The model may inadequately perform in certain demographic groups, exacerbating inequities with detection. Missing demographic data also created a limitation when assessing performance across populations.
3. **Model Complexity**: While EfficientNetV2B0 is resource-efficient, it may miss subtle abnormalities in complex mammograms, leading to lower performance in difficult cases.
4. **Handling of Missing Data**: Forward and backward imputation may overlook hidden patterns in non-random missing data, potentially impacting accuracy where missing values carry clinical significance.
5. **Feature Engineering**: Simplifying the feature space via dimensionality reduction may have glossed over any domain-specific features or advanced image-processing methods that may have improved performance.
6. **Ethical Risks in Deployment**: False negatives risk delaying diagnoses and treatment, while false positives could cause unnecessary distress and costs. For clinical deployment, balancing sensitivity and specificity and ensuring continuous validation are paramount.

Stage 7: Model Implementation

Our breast cancer detection model is implemented using TF Lite for deployment, optimizing it for resource-constrained devices while enabling real-time inference. Key steps in the implementation process include:

1. **Model Saving:** The model is serialized as a .pkl file to ensure compatibility with the specified dependencies in requirements.txt.
2. **Deployment Environment:**
    - **Operating System:** Ubuntu 22.04.3 LTS
    - **Python Version:** 3.10.12
    - **Dependencies:** Listed in the requirements.txt file.
3. **Inference:** The model supports real-time inference to provide immediate predictions, facilitating quick clinical decisions during patient consultations.
4. **Scalability:** While initally being deployed for real-time single-patient predictions, our model architecture allows scaling to handle larger workloads as needed.

Stage 8: Model Monitoring

To maintain reliable performance in a clinical setting, the following metrics will be continuously monitored:

- **Precision:** Ensures minimal false positives.
- **Recall:** Critical for capturing all potential malignancies.
- **F1 Score:** A balanced metric for precision and recall.
- **Precision-Recall AUC:** Provides robust evaluation for imbalanced datasets.
- **Inference Time:** Targets ≤ 200 ms for real-time results.
- **False Negative Rate (FNR): T**racks missed detections, prioritizing patient safety.
- **Model Drift:** Monitors deviations in input data or performance metrics.

Thresholds:

- **Green:** Optimal performance.
- **Yellow:** Declining performance— investigate and retrain if necessary.
- **Red:** Critical performance issues— decommission the model and revert to a backup system.

Stage 9: Scheduled Reviews

Conducting thorough and regular reviews will ensure the model remains effective and aligned with evolving clinical practices:

1. **Review Frequency:**

- **Routine Reviews:** Every 6–12 months to incorporate updated data and check for drift.
- **Event-Triggered Reviews:** This will be initiated if significant drift, performance degradation, or new imaging technologies arise.

2. **Review Process:**
   - **Evaluate metrics** against predefined thresholds.
   - **Retrain the model** on new data if drift or declining performance is seen.
   - **Validate updates** with expert radiologists and curated test datasets.
   - **Ensure compliance** with medical and regulatory standards.

Stage 10: Model Decommission

The model will be decommissioned when it fails to meet performance thresholds or becomes outdated due to new advancements in imaging or diagnostics. Steps of action include:

1. **Criteria for Decommission:**
   - Continuous red flags in precision, recall, or FNR.
   - Unrecoverable model drift or concept drift.
   - Introduction of superior models or technologies.
2. **Decommission Process:**
   - **Pull the Model:** Remove the model from production promptly.
   - **Fallback Mechanism:** Implement a manual review system or backup model to maintain clinical workflow.
   - **Performance Audit:** Analyze the root causes for failure to implement in future improvements.
   - **Model Retirement:** Archive the model, training data, and performance logs for accountability and version control.
3. **Documentation:** Maintain clear records of the decommissioning process, including performance issues, decisions made, and compliance with any regulatory standards.