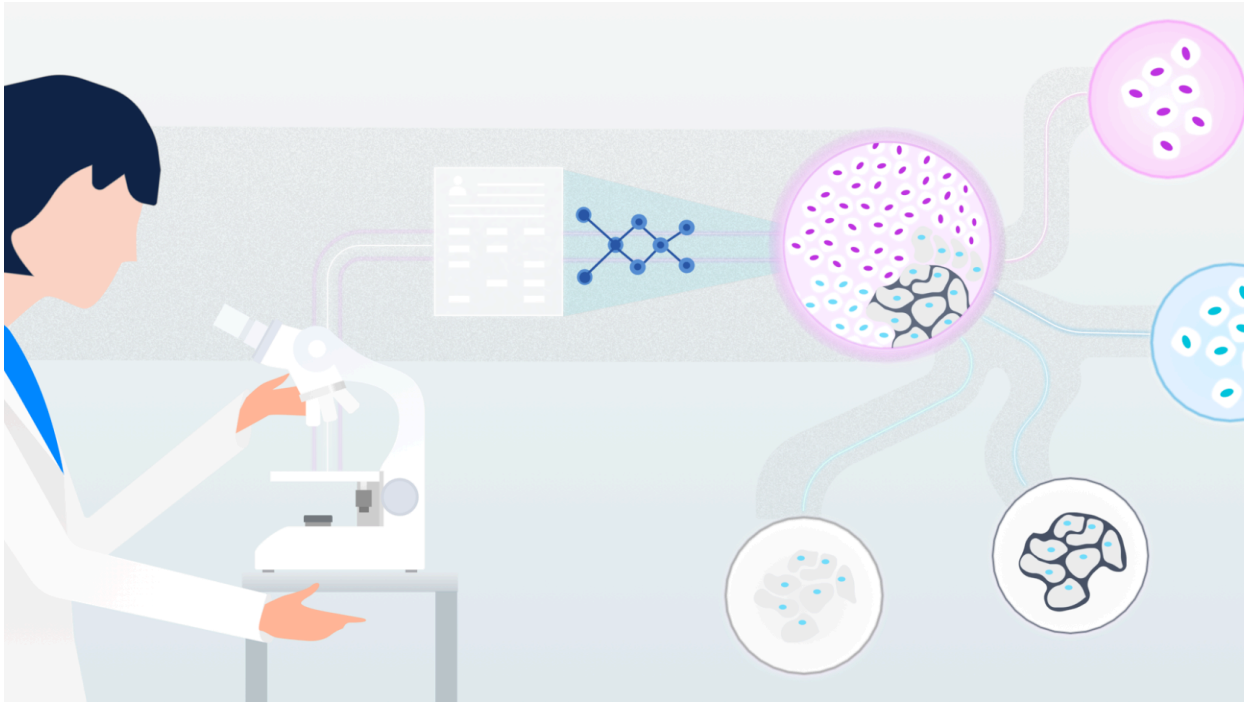
Community Engineering Project

Breast Cancer Detection using Machine learning



Author

Gupta Prashant Awadhesh Kumar (SY-B-17)

SOE23201010438, Pimpri Chinchwad University, Pune

Guided by

Dr. Sachin Jadhav

Abstract

Breast cancer is a leading cause of mortality among women globally. Early and accurate detection significantly enhances treatment outcomes and survival rates. Traditional diagnostic methods, such as mammography, have been instrumental in screening programs; however, they are not without limitations, including false positives and negatives. In recent years, the integration of machine learning (ML) techniques into medical diagnostics has shown promise in enhancing the accuracy and efficiency of breast cancer detection. This report provides a comprehensive overview of the application of ML in breast cancer detection, examining various modalities and discussing the challenges and future directions in this evolving field.

Introduction

Breast cancer remains one of the most prevalent cancers affecting mens & womens worldwide, with significant implications for public health. According to the World Health Organization (WHO), early detection through effective screening programs is crucial for reducing mortality rates associated with this disease. Traditional diagnostic methods, particularly mammography, have been the cornerstone of breast cancer screening. While mammography has contributed substantially to early detection, it is not without limitations. False positives can lead to unnecessary biopsies and anxiety, while false negatives may delay critical treatment. These challenges underscore the need for more accurate and reliable diagnostic tools.

In recent years, the advent of machine learning (ML) has opened new avenues in medical diagnostics. ML, a subset of artificial intelligence, involves algorithms that can learn from and make predictions based on data. In the context of breast cancer detection, ML algorithms have the potential to analyze complex patterns and patient data, thereby assisting clinicians in making more informed decisions. The application of ML in this domain aims to enhance diagnostic accuracy, reduce human error, and ultimately improve patient outcomes.

Literature Review

Breast cancer is a common and serious disease. Doctors use mammograms (X-ray images of the breast) to find cancer early, but these tests are not always correct. Sometimes, they say a person has cancer when they don't (false positive), or they miss cancer when it is there (false negative). To improve this, researchers have started using **machine learning (ML)**, a type of machine intelligence that helps computers learn from data and make predictions. Many studies show that ML can help detect breast cancer more accurately than traditional methods. Some common ML techniques used for this are **Logistic Regression, Decision Trees, and Random Forest**. These methods help doctors analyze patterns in medical data and make better decisions about whether a tumor is cancerous or not.

Identified Gaps and Project Contribution:

Despite these advancements, several gaps remain in existing research and implementations. Many ML-based breast cancer detection systems rely heavily on imaging techniques, often requiring extensive labeled datasets that are not always available.

Our project addresses these gaps by:

1. Implementing a **Flask-based web application** that enables users to manually enter features or upload a PDF for automated feature extraction, increasing accessibility and usability.
2. Utilizing **multiple ML models** (Logistic Regression, Decision Tree, and Random Forest) to improve prediction accuracy and reduce biases in individual models.
3. Applying **feature scaling and preprocessing techniques** to enhance model generalizability and robustness, mitigating the risk of overfitting.
4. Providing an efficient **decision-support tool** for early and accurate breast cancer detection, helping clinicians make informed diagnoses.

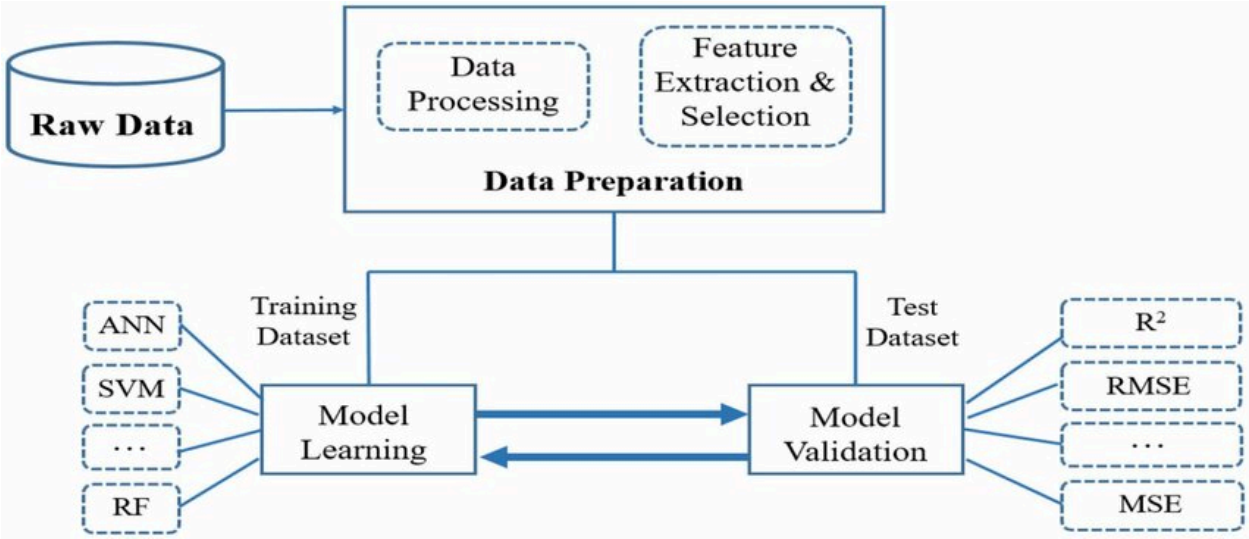
By integrating ML with a web-based interface, this project enhances breast cancer detection accessibility and contributes to advancing AI-assisted medical diagnostics.

Methodology: ML Algorithms

In the realm of breast cancer detection, a variety of machine learning algorithms have been employed to enhance diagnostic accuracy and efficiency. **Support Vector Machines (SVM)** classify data by identifying the optimal hyperplane that separates different classes, effectively distinguishing between malignant and benign cases. **Random Forest algorithms** construct multiple decision trees during training, offering robustness and the ability to handle complex datasets, thereby improving prediction reliability. **Regression algorithms**, particularly **Logistic Regression**, model the probability of a binary outcome to differentiate between benign and malignant tumors. The **algorithm** has been utilized for its simplicity and effectiveness in classifying breast cancer stages by analyzing the proximity of data points to predefined categories. These diverse algorithms collectively contribute to more accurate and efficient methodologies for breast cancer detection.

ML algorithms used in the project for breast cancer detection:

- **Logistic Regression:** Utilized to estimate the probability of a tumor being malignant or benign.
- **Decision Tree Classifier:** Uses an entropy-based approach to split the dataset into classes.
- **Random Forest Classifier:** Combines multiple decision trees to reduce overfitting and improve prediction accuracy.



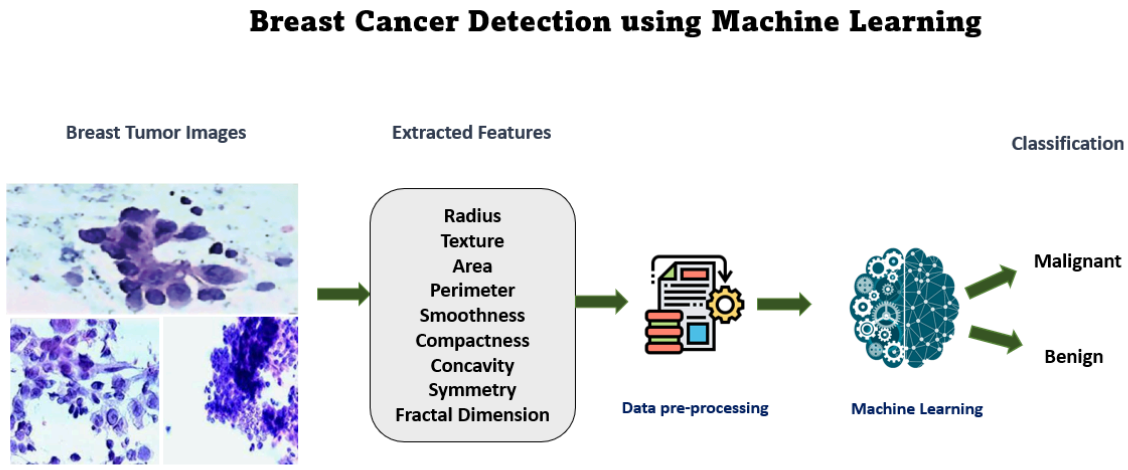
Methodology: Data Preprocessing and Model Training

Data preprocessing is crucial for effective model training. In our project, the following steps were performed:

- **Exploratory Data Analysis (EDA):**
  • Dataset shape and statistical summaries were reviewed.
  • Missing values were handled by dropping columns with null entries.

- **Encoding and Scaling:**
  • The diagnosis labels were encoded (e.g., Malignant as 1 and Benign as 0).
  • Feature scaling was applied using StandardScaler to ensure that data dimensions were normalized.

- **Train-Test Split:**
  • Data was divided into training (80%) and test (20%) sets for model evaluation.



Methodology: Prediction Model & Tumor Features

The breast cancer detection system uses machine learning algorithms designed to differentiate between malignant and benign tumors using 29 clinically significant features. The model undergoes feature extraction, and selection processes to ensure that only the most relevant information is used to support an accurate and early diagnosis.

The 29 Features and Their Roles in Prediction

1. **Mean Radius:** Average radius of tumor cell nuclei; a primary indicator of tumor size.
2. **Mean Texture:** Measures the variation in gray-scale intensity, reflecting tissue heterogeneity.
3. **Mean Perimeter:** Helps in understanding the boundary irregularity of the tumor.
4. **Mean Area:** Indicates the overall size of the lesion, correlating with potential malignancy.
5. **Mean Smoothness:** Quantifies the smoothness of the tumor surface; irregular surfaces may indicate malignancy.
6. **Mean Compactness:** Ratio of perimeter squared to area; a measure of shape irregularity.
7. **Mean Concavity:** Assesses the severity of concave portions of the tumor's contour.
8. **Mean Concave Points:** Count of concave segments in the boundary, highlighting irregular growth.
9. **Mean Symmetry:** Reflects the bilateral symmetry of the tumor; lower symmetry can be suspicious.
10. **Mean Fractal Dimension:** Describes the complexity of the tumor boundary.
11. **Radius Error:** Standard deviation of the radius, highlighting inconsistencies in size.
12. **Texture Error:** Variation in texture measurement, which may indicate heterogeneity.
13. **Perimeter Error:** Standard deviation in the perimeter measurement, underscoring boundary irregularity.
14. **Area Error:** Error in area measurement that can signal abnormal growth patterns.
15. **Smoothness Error:** Variation in smoothness; greater error might reflect erratic tumor growth.

16. **Compactness Error:** Variability in compactness, providing insight into shape anomalies.
17. **Concavity Error:** Variability in concavity measurements, indicating irregular cell formations.
18. **Concave Points Error:** Error in counting concave points, reinforcing boundary irregularity clues.
19. **Symmetry Error:** Standard deviation in symmetry, helping to quantify abnormal growth.
20. **Fractal Dimension Error:** Variability in the fractal dimension, reflecting complex structural variations.
21. **Worst Radius:** Maximum radius observed, often linked with aggressive tumor behavior.
22. **Worst Texture:** Highest texture value recorded, underscoring heterogeneous tissue composition.
23. **Worst Perimeter:** Largest measured perimeter, further emphasizing tumor irregularity.
24. **Worst Area:** Maximum lesion area, often indicative of malignant growth.
25. **Worst Smoothness:** Peak irregularity in tumor surface texture.
26. **Worst Compactness:** Maximum compactness value, pointing to significant deviations from normal tissue.
27. **Worst Concavity:** Highest concavity score, reinforcing the presence of invasive tumor features.
28. **Worst Concave Points:** Peak count of concave points, strengthening the evidence for abnormal boundaries.
29. **Worst Symmetry:** Most pronounced asymmetry, a key marker in differentiating malignant tumors.

Utilization in the Prediction Model

This comprehensive approach not only increases detection accuracy but also enhances the interpretability of predictions, enabling clinicians to better understand the rationale behind each diagnosis. Ultimately, our model serves as a decision support tool, facilitating early and reliable breast cancer detection.

Implementation: Model Development Code Explanation

A key part of the project involves training multiple classifiers. Below is an snippet from the `breast_cancer_detection.py` file, which defines the `models` function:

```
def models(X_train, Y_train):

    # Logistic regression classifier

    log = LogisticRegression(random_state=0)

    log.fit(X_train, Y_train)

    # Decision tree classifier

    tree = DecisionTreeClassifier(criterion='entropy', random_state=0)

    tree.fit(X_train, Y_train)

    # Random forest classifier

    forest = RandomForestClassifier(n_estimators=10, criterion='entropy',
random_state=0)

    forest.fit(X_train, Y_train)

    # Print the accuracy of each model on the training dataset

    print("The accuracy of Logistic Regression: ", log.score(X_train,
Y_train))

    print("The accuracy of Decision Tree: ", tree.score(X_train, Y_train))

    print("The accuracy of Random Forest: ", forest.score(X_train, Y_train))

    return log, tree, forest
```

Accuracy of Logistic Regression: 98%

Accuracy of Decision Tree: 100%

Accuracy of Random forest: 99%

Implementation: Flask Based Application

The project integrates a Flask web application to enable user-friendly interaction. The application accepts both manual data entry and PDF uploads for feature extraction. Below is an important segment from `app.py` that handles the prediction request:

```
@app.route("/predict", methods=["POST"])
def predict():
    try:
        if "pdf_file" in request.files and
request.files["pdf_file"].filename != "":
            pdf_file = request.files["pdf_file"]
            input_features = extract_features_from_pdf(pdf_file)
        else:
            input_features = [float(request.form[f]) for f in features]
        input_array = np.array(input_features).reshape(1, -1)
        prediction = model.predict(input_array)[0]
        result = "Malignant (Cancerous)" if prediction == 1 else
"Benign (Non-Cancerous)"
        return jsonify({"prediction_text": f"Prediction: {result}"})
    except Exception as e:
        return jsonify({"prediction_text": f"Error: {str(e)}"})
```

Explanation:

This route handles POST requests for prediction. Depending on whether a PDF file is provided, it either:

- Extracts feature values from the PDF using a helper function.
- Or, gathers manual input from form fields.

The features are reshaped and passed to the ML model for prediction, returning the result as a JSON response.

Implementation: PDF Feature Extraction Method

The following helper function from `app.py` extracts the 29 required feature values from an uploaded PDF file:

```
def extract_features_from_pdf(pdf_file):
    doc = fitz.open(stream=pdf_file.read(), filetype="pdf")
    text = ""
    for page in doc:
        text += page.get_text("text") + "\n"
    extracted_data = {}
    for feature in features:
        pattern = rf"({feature})\s*[:\-]?\s*([\d\.]+)"
        match = re.search(pattern, text, re.IGNORECASE)
        if match:
            extracted_data[feature] = float(match.group(2))
        else:
            raise ValueError("Could not extract all required features from the PDF.")
    return [extracted_data[feature] for feature in features]
```

Explanation:

- **PDF Processing:** Uses PyMuPDF (fitz) to read text from each page of the PDF.
- **Regex Matching:** For each feature, a regular expression searches the text for numeric values.
- **Validation:** If any required feature is missing, an error is raised to ensure data integrity.

This method automates feature extraction, allowing the ML model to process PDF uploads seamlessly.

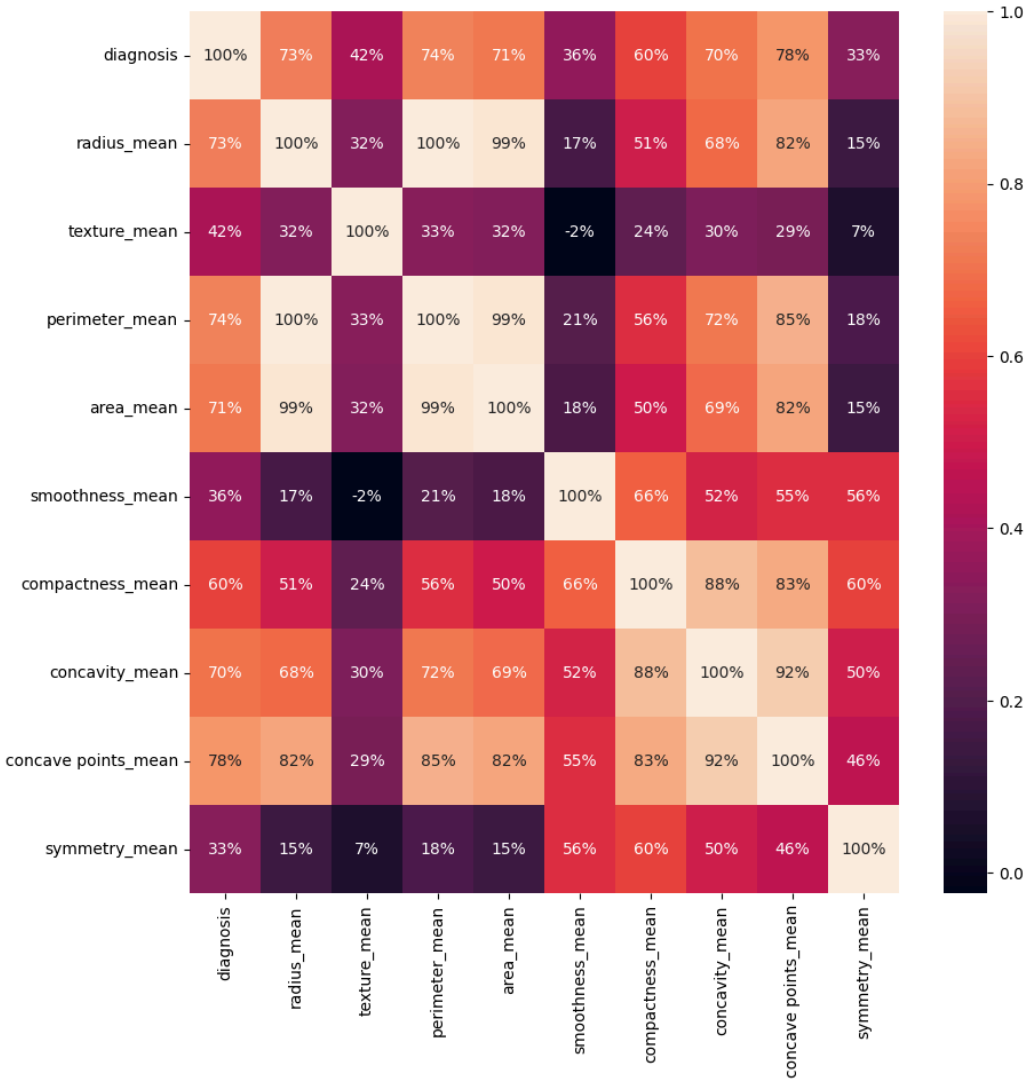
Result and Performance

After model training, the performance was evaluated using accuracy metrics and confusion matrices. The key performance figures obtained were:

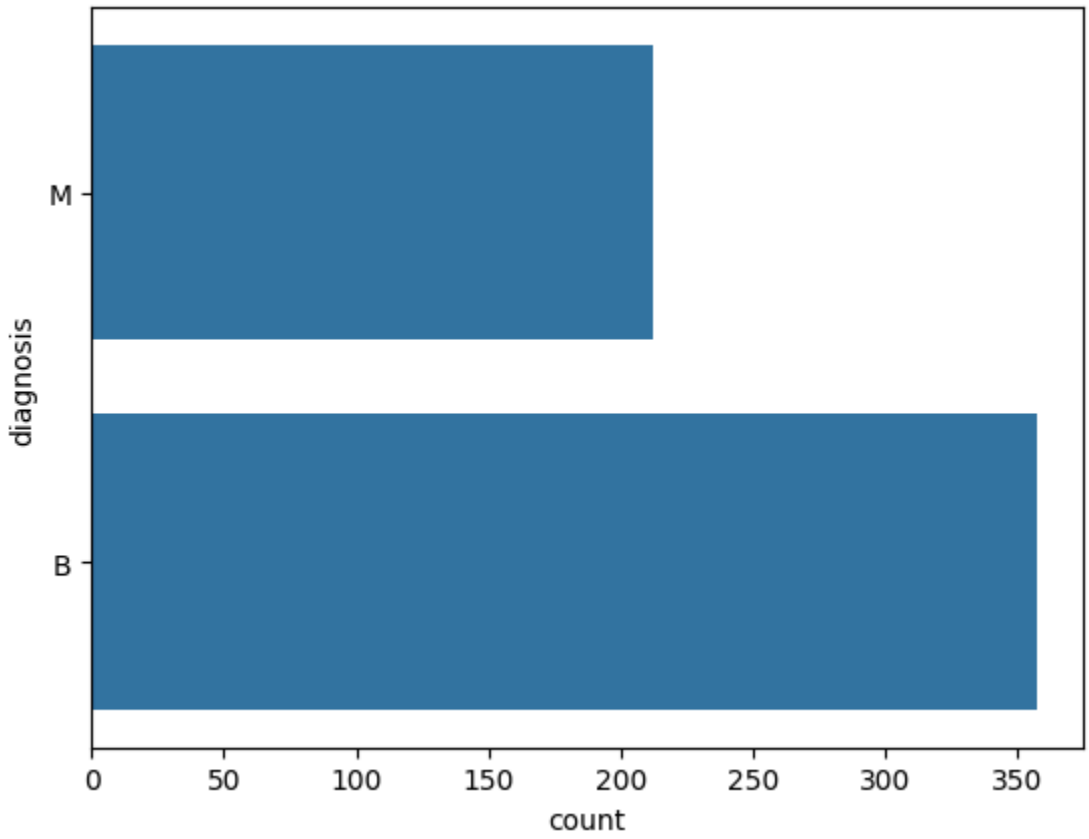
- **Logistic Regression:** ~98.90% accuracy
- **Decision Tree:** 100% accuracy on the training set (with potential overfitting)
- **Random Forest:** ~99.78% accuracy

The confusion matrix and classification reports confirm that the models can effectively distinguish between malignant and benign tumors.

Confusion matrix heatmap:



Performance metrics graph:



Conclusion: Impact and Benefits

The integration of machine learning in breast cancer detection offers transformative benefits across multiple facets of healthcare:

Enhanced Diagnostic Accuracy:

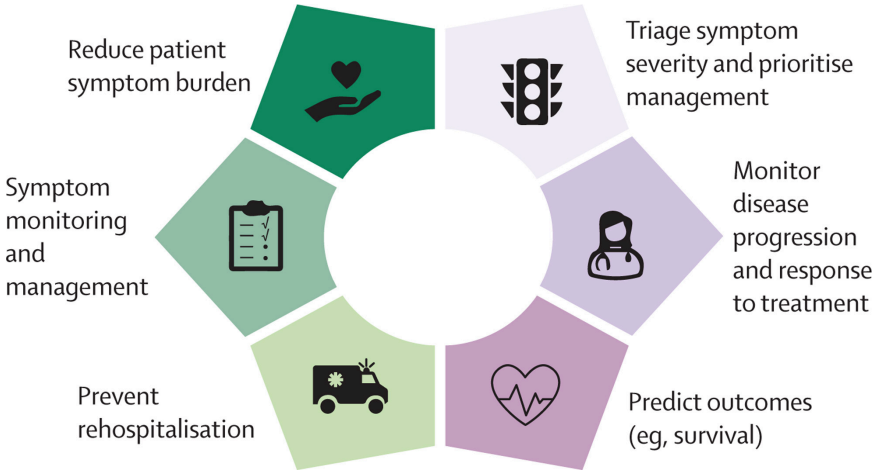
Machine learning algorithms excel at detecting subtle patterns within medical reports. By analyzing large, complex datasets, these models reduce human error and offer a level of precision that significantly improves diagnostic reliability. Their ability to process and learn from diverse data points enables early detection of malignant tumors, which is critical for effective treatment planning.

Timely Interventions:

Early detection is paramount in the battle against breast cancer, as it allows for the prompt initiation of treatment protocols. ML models provide rapid assessments of report data, shortening the time and avoiding the errors. This rapid turnaround facilitates quicker clinical decision-making, ensuring that patients receive appropriate interventions at the earliest possible stage.

Resource Optimization:

Automated analysis through machine learning significantly eases the workload on radiologists and medical staff. ML systems free up valuable time for healthcare professionals to focus on complex cases and patient care. This streamlined process not only increases efficiency within clinical workflows but also reduces operational costs.



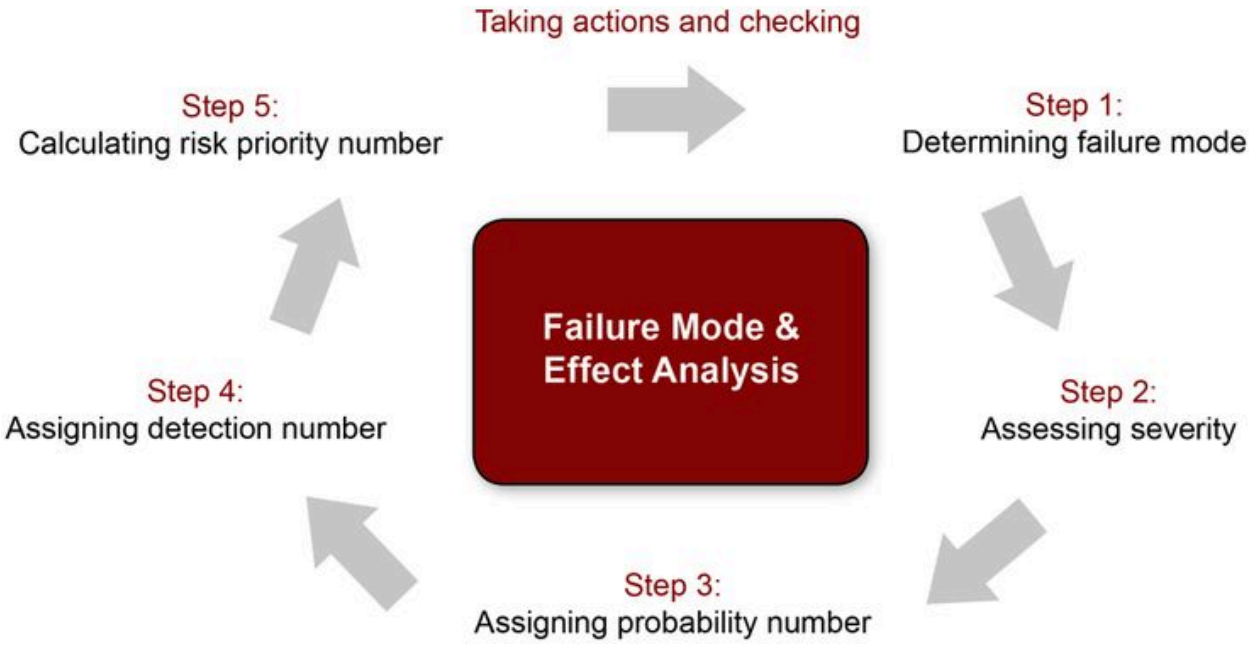
Conclusion: Discussion of Challenges and Limitations

The project's challenges include:

- **Data Extraction Issues:**
  Ensuring all 29 features are accurately extracted from diverse PDF formats can be error-prone. Missing or misinterpreted data may lead to prediction errors.

- **Overfitting Concerns:**
  The high training accuracy, especially for the Decision Tree, may indicate overfitting. Additional validation and cross-validation are necessary to ensure generalizability.

- **Integration Complexity:**
  Seamless integration between the ML backend and the Flask web interface requires robust error handling and user feedback mechanisms.



Scientific Terms of the Topic

Keywords: Mammogram, Mortality Rate, SVM, Malignant Tumor, Benign Tumor, Histopathology, Feature Scaling, Overfitting

- **Mammogram:** An X-ray image of the breast used for early detection of cancer.
- **Mortality Rate:** The number of deaths in a population during a specific period.
- **Support Vector Machine (SVM):** An algorithm that finds the optimal hyperplane to classify data points into distinct classes.
- **Malignant Tumor:** A cancerous growth that has the potential to invade surrounding tissues and spread.
- **Benign Tumor:** A non-cancerous growth that typically grows slowly and remains localized.
- **Histopathology:** The study and diagnosis of disease through the examination of tissue samples under a microscope.
- **Feature Scaling:** A method to normalize the range of independent variables in data processing.
- **Overfitting:** A modeling error where a model is excessively complex and performs well on training data but poorly on unseen data.

References

Udemy: Breast Cancer Detection Using Machine Learning

Research work: Research Gate, IEEE Xplore, SciHub

Images: Google Images