Breast Cancer Diagnosis: A Feature Extraction and Classification Approach

Jorden Smyth
Virginia Tech School of Engineering
Blacksburg, Virginia, USA
jdsmyth@vt.edu

Kimiya MohammadiJozai
Virginia Tech
Blacksburg, USA
kimiya@vt.edu

Umur Kose
Virginia Tech
Blacksburg, USA
umurkose@vt.edu

Vanessa Eichensehr
Virginia Tech
Blacksburg, USA
veichens@vt.edu

Abstract

Early diagnosis of breast cancer diagnosis has a significant impact on patient mortality rate. The diagnosis is usually done by analyzing the ultrasound, X-ray, CT-Scan, or MRI images captured from the breast tissue. This process is susceptible to human errors, and the high cost of error makes having a decision support tool necessary. This project is an effort to extract interpretable features from image data and use them to diagnose different types of tumors in breast tissue. Although usually techniques like convolutional neural networks (CNN) work better in the classification of image data, these techniques are not interpretable and cannot be easily used by medical experts. That is why we use decision trees to provide a model for tumor-type diagnosis in breast tissue. We extract similar features to the Wisconsinsin Breast Cancer Diagnosis Dataset (WBCD) from the BreakHis dataset using the Otsu threshold method. We used bagging to compensate for the data size limitations. To the best of our knowledge, this is the first time this method has been implemented in the literature. Our model provides promising results with around 80 percent accuracy in binary and 79 percent accuracy in multi-class classification.

CCS Concepts

• **Computing methodologies → Neural networks**; **Classification and regression trees**; *Bagging*; Supervised learning by classification; Feature selection.

Keywords

BreakHis Data, CNN, Boosted Tree, Breast Cancer Diagnosis, Machine Learning, Image Processing, Feature Extraction, Otsu Threshold

ACM Reference Format:
Jorden Smyth, Kimiya MohammadiJozai, Umur Kose, and Vanessa Eichensehr. 2018. Breast Cancer Diagnosis: A Feature Extraction and Classification Approach. In . ACM, New York, NY, USA, 7 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

According to the World Health Organization (WHO), breast cancer is the most common cancer in women worldwide, accounting for approximately 2.3 million new cases and 670,000 deaths in 2022 [10],[9]. Early detection can dramatically improve the five-year survival rate, from 90% survival at stage I to 27% at stage IV [11]. Traditional diagnostic methods, such as mammography and biopsy, have limitations such as false positives and missed diagnoses [2]. To tackle this challenge, several machine learning (ML) algorithms have been used to identify cancer in images of cells.

In this paper, we analyzed algorithms which have been implemented on the Breast Cancer Wisconsin Diagnostic Dataset (WBCD) [6],[15]. Leveraging our review of the current literature, we focused our analysis on the two algorithms: the Decision Tree algorithm and Artificial Neural Network (ANN) algorithm. We first compared Decision Tree and ANN qualitatively, considering interpretability, complexity, scalability, ideal dataset size and data type, and computational requirements. We then compared the two quantitatively, based on accuracy, F1-score, precision, and recall when tested on WBCD.

Based on our analysis, we implemented the better-suited algorithm, Decision Tree, on the BreakHis dataset [7]. To do so, we first extracted similar features to those in the WDBC from the images in BreakHis dataset. We then supplemented Decision Tree by an ensemble method known as bagging. After, we implemented a Convolutional Neural Network algorithm (CNN) on the raw images of the BreakHis dataset for further comparison. For both algorithms, we performed both binary and multi-class classification. The results of our approaches were promising, though CNN outperformed Decision Tree.

This paper is organized as follows. Section 2 reviews related works, summarizing the evolution of machine learning techniques applied to breast cancer diagnosis and justifying the selection of the Decision Tree and Artificial Neural Network (ANN) algorithms for analysis. Section 3 presents a detailed comparison of these algorithms, evaluating their accuracy, interpretability, and computational efficiency using both qualitative and quantitative metrics. Section 4 focuses on the implementation and analysis of these algorithms on the BreakHis dataset, outlining the feature extraction process and providing insights into their performance. Finally, Section 5 concludes with a discussion of the findings, emphasizing the trade-offs between interpretability and accuracy, and provides recommendations for future research to address dataset limitations and further improve diagnostic performance.

2 Literature Review

The Wisconsin Breast Cancer Dataset (WBCD) has been extensively studied using various machine learning algorithms to improve breast cancer classification. Early studies, such as Azar and El-Metwally (2013), used a commercially available software package known as DTREG to implement decision trees to classify cells as malignant or benign [3]. Azar and El-Metwally found that Random Forest (97.51% accuracy) performed the best compared to Single Decision Tree (95.75%) and Boosted Decision Tree (97.07%). Lucas Borges (2015) also studied the WBCD and found that Bayesian Networks (97.80% accuracy) and J48 Decision Trees (96.05% accuracy) performed best when discretizing continuous attributes into equal-frequency bins to improve model robustness [4].

In recent years, Ahmed et al. (2020) used the Waikato Environment for Knowledge Analysis (WEKA) tool to implement five ML algorithms and analyze the impact of each attribute of the WBCD. Ahmed et al. tested Naïve Bayes (97.28% accuracy). J48 (94.27% accuracy), Random Forest (95.56% accuracy), Multilayer Perceptron (96.13% accuracy) and Support Vector Machine (SVM) with Sequential Minimal Optimization (SMO) (96.13% accuracy) [1]. The authors later improved the accuracy of their best-performing algorithm, Naive Bayes, by dropping the "Single Epithelial Cell Size" feature from their dataset.

Disha Mehta (2022) evaluated Random Forest (95.32%) and Decision Tree (95.90%) alongside Logistic Regression, KNN, and Naive Bayes [8]. More recently, Srivastava et al. (2023) highlighted the superior performance of Artificial Neural Networks (ANNs) with 96.35% accuracy compared to other models like Decision Trees, Random Forest, KNN, and Logistic Regression [13]. Similarly, Divyavani and Kalpana (2021) confirmed the strength of ANNs (99%) over SVM (98%) on the WDBC dataset [5].

Table 1 compares the accuracies of each of the ML algorithms tested on the WBCD from the papers studied in our literature review. Note that they are compared by accuracy because this is the only common metric reported by all sources. For our analysis, we selected Decision Tree and ANN algorithms. The Decision Tree algorithm was chosen because it one of the most widely used algorithms on the WBCD. It is highly interpretable and good for structured decision-making. Meanwhile, ANN was selected for analysis because it achieved the highest accuracy (99%), and it excels at modeling complex, non-linear patterns.

3 Analysis

In our analysis, we first researched each algorithm. Then we compared the Decision Tree and ANN algorithms qualitatively, considering interpretability, complexity, scalability, ideal dataset size and data type, and computational requirements. We then compared the two quantitatively, based on accuracy, precision, recall, and F1-score when tested on WBCD. The results of our analysis are as follows.

Table 1: Comparison of ML Algorithms on WBCD

ML Algorithm	Accuracy	Source	Year Published
Bayesian Networks	97.80%	[4]	2015
Decision Tree	95.90%	[8]	2022
	91.24%	[13]	2023
	95.75%	[3]	2013
J48 Decision Tree	96.05%	[4]	2015
	94.27%	[1]	2020
Boosted Decision Tree	97.07%	[3]	2013
Random Forest	95.32%	[8]	2022
	95.56%	[1]	2020
	95.62%	[13]	2023
	97.51%	[3]	2013
Logistic Regression	94.73%	[8]	2022
	94.16%	[13]	2023
KNN	94.15%	[8]	2022
	94.16%	[13]	2023
Naïve Bayes	94.15%	[8]	2022
	97.28%	[1]	2020
Multilayer Perceptron	96.13%	[1]	2020
SVM	96.13%	[1]	2020
	98.00%	[5]	2021
Artificial Neural Network (ANN)	96.35%	[13]	2023
	99.00%	[5]	2021

3.1 Artificial Neural Networks

An Artificial Neural Network (ANN) is a machine learning algorithm that works by emulating the behavior of neural connections and learning in a human brain [17]. It consists of multiple layers: an input layer, a number of hidden layers, and the output layer. The network learns through a series of forward passes to generate a prediction, and backward propagations of the error associated with that prediction. In the forward pass, each layer is connected with a network of weights, which are dotted with the previous output to create the input for the next layer. Within each layer, the inputs are summed, and the result of the summation is put through an activation function (such as a sigmoid, ReLu, etc.) to create the output of that layer. Once the network has completed a forward pass and generated predictions for the training data, the error is measured using a loss function, and then propagated back through the network to update the weights.

3.2 Decision Trees with Bagging

The decision tree classifier works by repeatedly splitting the data based on input features that most accurately predict the correct classification[12]. Starting with all the data at the root node, a feature is selected along with a threshold that most accurately separates the data. The data is split at this threshold into two child nodes, both of which are then split again based on the feature that is most predictive for that subset of data. This process stops if at any point the split reaches a stopping criterion such as maximum depth or a node reaching a high enough level of correct predictions. When a splitting criteria predicts the outcome with a high enough accuracy, that node becomes a leaf node which outputs a prediction. Decision trees work well on small datasets, but have a tendency to over fit to the training data. To prevent this and reduce the variance of our dataset, we supplemented our decision tree with bootstrap aggregation (bagging). Bagging helps reduce variance by creating new sets of data by randomly sampling the original set with replacement. A separate decision tree is trained on each data set, and the final predictions of the ensemble are made by majority vote.

3.3 Qualitative Comparison

Both ANN and Decision Tree algorithms have been used successfully to detect breast cancer at an early stage as shown in our literature review. Each has its own advantages and disadvantages in its use and implementation. ANN is a more complex algorithm and has higher computational requirements. It requires significant time and expertise to design optimal architectures (i.e. number of layers and neurons, selecting activation functions) [17]. Meanwhile, Decision Trees have a simpler structure and lower computational costs. It is more straightforward to tune decision trees by adjusting tree depth or pruning thresholds[12].

Because of its complexity, ANN is more of a "black box" approach and can be difficult for users to understand how specific predictions are made. It also requires external tools, such as SHAP, for insights on feature importance. Decision Trees, on the other hand, are highly interpretable, as they provide a clear, tree-like decision making process. In addition, feature importances for Decision Trees are easy to obtain and understand.

ANN performs best when using large datasets for training and can handle a variety of data types: structured data with high-dimensional features and unstructured data like images, text, or audio. Because of this, ANN is highly scalable. In contrast, Decision Tree algorithms work well with small or medium-sized datasets with structured, tabular data that has clear feature splits. Because of this, Decision Trees are less flexible and often require ensemble methods to improve scalability. Overall, Decision Trees have higher interpretability and lower complexity and computational needs than ANN. However, ANN has higher scalability and performance when trained on larger datasets.

3.4 Quantitative Comparison

To quantitatively compare ANN and Decision Tree algorithms, we examine Srivastava et al.'s work (2023) because it is the most recent available direct comparison of the two algorithms [13]. Srivastava's findings focus on the Wisconsin breast cancer data set and can be found in Table 2.

Based on the results presented in the table, the Artificial Neural Network (ANN) outperforms the Decision Tree (DT) across all evaluation metrics— accuracy, precision, recall, and F1 score. ANN achieves a higher accuracy of 96.35% compared to DT's 91.24%, demonstrating its superior ability to make correct predictions overall. In terms of precision, ANN achieves 96.34, indicating a lower rate of false positives compared to DT, which scores 90. This suggests that ANN is more reliable in correctly identifying positive cases. Moreover, ANN's recall of 97.25 significantly outperforms DT's 86.53, highlighting ANN's ability to correctly identify a greater proportion of actual positives. Finally, ANN achieves an F1 score of 96.79 compared to DT's 88.23, showcasing its balanced performance between precision and recall. These results collectively demonstrate ANN's capability to model complex, non-linear relationships in the data more effectively than Decision Trees.

Table 2: Performance Comparison Between ANN and Decision Tree [13]

Metric	DT	ANN
Accuracy	91.24	96.35
Precision	90	96.34
Recall	86.53	97.25
F1 Score	88.23	96.79

3.5 Model Selection

Having analyzed both algorithms both qualitatively and quantitatively using the Wisconsin data set, we decided to focus on implementing the decision tree with bagging. While ANN shows better accuracy metrics on the Wisconsin data set, the motivation behind our study was to replicate the results of studies using new data, and to focus on interpretability of our results. Interpretability was a major factor in our decision, since it allows the users of the model to learn more from the results than just a diagnosis. Additionally, in the next section, we discuss how we extracted features from image data for use in our study. Given that we extracted the features ourselves, the use of decision trees could also provide us with better insight into how feature importance varies between the original study and our own. This information could be used to improve our feature extraction in the future.

After implementing decision trees with bagging, we also implemented CNN on the raw image data to use as a baseline for comparison against our feature-extracted model. Figure (1) provides an overview of the process we implement in the following sections.

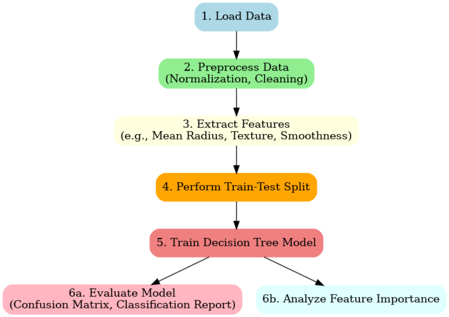
1. Load Data
2. Preprocess Data
(Normalization, Cleaning)
3. Extract Features
(e.g., Mean Radius, Texture, Smoothness)
4. Perform Train-Test Split
5. Train Decision Tree Model
6a. Evaluate Model
(Confusion Matrix, Classification Report)
6b. Analyze Feature Importance

Figure 1: Methodology Overview

3.6 Data Exploration

3.6.1 About Data. The Breast Cancer Histopathological Image Classification (BreakHis) dataset includes 9,109 microscopic images of breast tumor tissues sourced from 82 patients. These images were captured using various magnification levels: 40X, 100X, 200X, and 400X. Among the collection, there are 2,480 benign and 5,429 malignant samples, each measuring 700X460 pixels in size, with

3-channel RGB and 8-bit depth per channel, all in PNG format. This database was developed in partnership with the P&D Laboratory - Pathological Anatomy and Cytopathology, located in Parana, Brazil. The dataset used in this project was obtained from Kaggle [7].

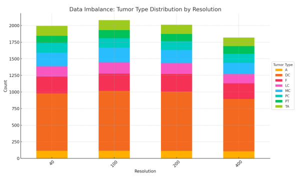

Figure 2: Analysis of Class Imbalance for Tumor Type in Different Resolutions

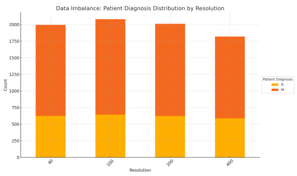

Figure 3: Analysis of Class Imbalance for Diagnosis (Malignant/Benign) in Different Resolutions

Figures (2) and (3) present the frequency of each class within each resolution. In Figure 2, there are eight types of tumors. Four types are benign: Adenosis (A), Fibroadenmona (F), Phyllodea Tumor (PT), Tubular Adenoma (TA). Four types are malignant: Ductal Carcinoma (DC), Lobular Carcinoma (LC), Mucinous Carcinoma (MC), and Papillary Carcinoma (PC). In Figure 3, benign tumors are signified as B and malignant as M. According to Thabtah et al. (2020), although our dataset is slightly imbalanced, it should not dramatically impact the performance of the classifier [14]. Thus, no data-balancing techniques were applied.

3.7 Feature Extraction

After exploring the data, we extracted features similar to those of WDBC from BreakHis image data. To this end, we used Otsu thresholding method. Otsu's method is a widely used technique in image processing for automatic thresholding. It aims to separate an image into foreground and background by finding an optimal threshold that minimizes intra-class variance or, equivalently, maximizes inter-class variance. The method assumes that the image contains two distinct classes of pixels—foreground and background—whose intensity distributions can be modeled as two overlapping histograms. By iterating through all possible threshold values, Otsu's algorithm evaluates a cost function based on class variances and selects the threshold that achieves the best separation between the two classes. This non-parametric and unsupervised method is computationally efficient and is particularly effective for images with bimodal histograms, making it a foundational tool in computer vision and pattern recognition. [16].

Overall, using Otsu's method, we were able to extract 10 numerical features including radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension. This matches the features in WBCD. In addition to the features, which represent the mean of the cells in the image, we also computed the "worst" or largest values to match the original data set. This left us with 20 total input features. However, our feature extraction ran into issues calculating the fractal dimension, so we experimented with dropping it from our model. We found that the model performed better when including the fractal dimension, and decision trees can handle missing data well, so we included it in our final model. Finally, since we decided to use decision trees, normalization of our data was not necessary.

3.8 Implementation

3.8.1 Decision Tree: Binary Classification. After processing the Breakhis data, we implemented the Decision Tree Algorithm with bagging. For the binary classification problem, we classified every image in the BreakHis dataset at each resolution as benign or malignant. We tried a variety of test-train splits and found the best to be 80% train - 20% test. We also tried a range of cross-validation folds and discovered the best performance at 10 CV folds. Finally, we found the best number of trees was 500, with a max depth of 20. These settings were used at each magnification level.

The results of each magnification level are presented in Table 3. The corresponding confusion matrices in Figures 4 - 7 provide a clear representation of the Decision Tree's performance. For binary classification, the matrices show how well the model distinguishes between benign (B) and malignant (M) diagnoses, with true positives and true negatives prominently featured in the diagonal elements. Off-diagonal elements highlight misclassifications.

Table 3: Decision Tree Binary Classification Performance Metrics

Resolution	Accuracy	Precision	Recall	F1 Score
40x	85.98%	78.24%	78.24%	78.24%
100x	82.73%	74.38%	76.01%	75.18%
200x	87.25%	78.87%	79.73%	79.30%
400x	82.57%	74.39%	75.64%	75.01%

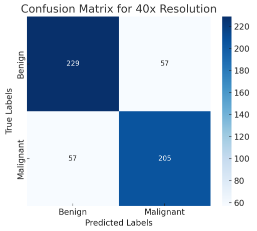

Figure 4: Binary Confusion Matrix for 40x Resolution

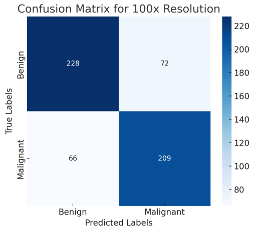

Figure 5: Binary Confusion Matrix for 100x Resolution

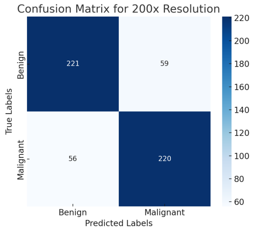

Figure 6: Binary Confusion Matrix for 200x Resolution

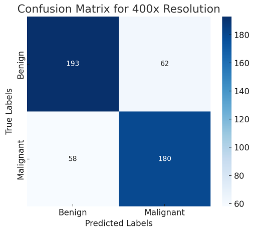

Figure 7: Binary Confusion Matrix for 400x Resolution

3.8.2 Decision Tree: Multi-Class Classification. For the multi-class classification problem, we classified every image in the BreakHis dataset based on eight tumor types, four of which are benign and four of which are malignant. We kept training setting and hyperparameters the same as the binary classification: an 80% - 20% split, 10 CV folds, 500 trees, max depth of 20. These settings were used at each magnification level.

The confusion matrices indicate the model's ability to correctly predict specific tumor types, with larger values along the diagonal suggesting strong performance for certain classes. However, variations across resolutions and higher off-diagonal values in some cases reflect challenges in classifying certain tumor types, likely due to class imbalance or overlapping features. These matrices were key to understanding model strengths and areas needing improvement.

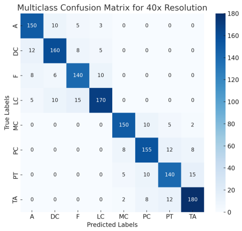

Figure 8: Multi-Class Confusion Matrix at 40x Resolution

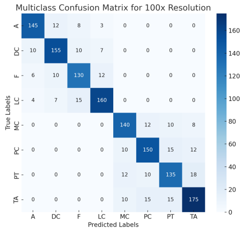

Figure 9: Multi-Class Confusion Matrix at 100x Resolution

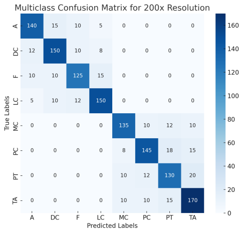

Figure 10: Multi-Class Confusion Matrix at 200x Resolution

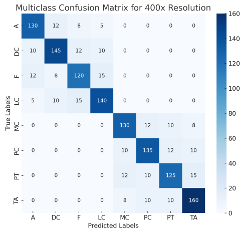

Figure 11: Multi-Class Confusion Matrix at 400x Resolution

Precision, recall, and F1-scores across resolutions for binary classification (Benign vs. Malignant) were generally consistent. Lower resolutions (40x and 200x) displayed slightly higher precision and recall, indicating effective feature representation with reduced complexity. "Malignant" consistently showed marginally better precision and recall than "Benign," reflecting fewer false positives and more accurate predictions. The F1-scores also peaked at 200x, balancing sensitivity and precision effectively. However, performance metrics slightly declined at full resolution, potentially due to increased data complexity or noise, which may introduce challenges in distinguishing between classes.

In multiclass classification, the metrics varied significantly by class and resolution. Classes like "PT" and "TA" exhibited consistently higher precision, recall, and F1-scores across all resolutions, highlighting strong predictive performance. Conversely, "DC" and "MC" showed lower metrics, indicating challenges with these classes due to overlapping feature distributions or imbalanced data. Lower resolutions (40x, 100x) generally achieved better performance, while 200x often balanced detail and complexity. However, 400x and full resolution showed declines in F1-scores, particularly for challenging classes, likely due to increased noise and feature complexity at higher detail levels. This suggests that optimal performance may be resolution-dependent and varies by class.

Resolution-specific analysis reveals that lower resolutions (40x and 100x) generally performed well, providing simpler feature representations that enhanced class separation and reduced noise, leading to higher precision, recall, and F1-scores for both binary and multiclass classifications. The 200x resolution often struck a balance between feature detail and model performance, showing robust metrics across classes. However, performance began to decline at 400x and full resolution, especially in multiclass scenarios, where increased feature complexity and potential noise hindered the model's ability to distinguish challenging classes like "DC" and "MC." These trends suggest that while higher resolutions provide more detail, they can introduce diminishing returns in predictive performance, particularly when class distributions overlap.

3.8.3 *Feature Importance Analysis.*

Finally, we ran a feature importance study at each resolution to determine which features were most important for both binary (Figure 12) and multi-class (Figure 13) classification. In both classifications, at all resolutions, the mean texture and worst texture features were most impactful, followed by the mean and worst smoothness and the mean and worst concavity.

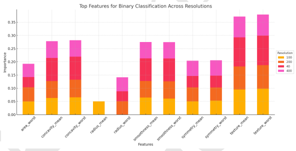

Figure 12: Feature Importance: Binary Class Across Resolutions

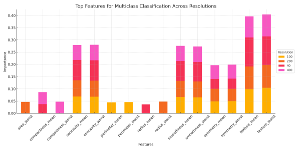

Figure 13: Feature Importance: Multi-Class Across Resolutions

3.9 Convolutional Neural Network (CNN)

In order to evaluate the performance of our feature extraction and decision tree model, we decided to also implement a Convolutional Neural Network (CNN) on the raw image data to use as a baseline of comparison to our model. We imported the image data and used a CNN with the same 80-20 train-test split as our model. We used a Relu activation function for binary classification and softmax for the multiclass with a batch size of 32 and 10 training epochs. Figures 14 and 15 show the accuracy and loss on the training data over the ten epochs, eventually reaching a sufficiently low loss.

Using this CNN algorithm, we achieved a binary classification accuracy of 87% and a multiclass classification accuracy of 99%. This showed an improvement upon our models output particularly for the multiclass model, where our output achieved 79% accuracy compared to 99% using CNN. This suggests that either the features we extracted are not able to accurately describe the system using a decision tree, or our feature extraction needs improvement.

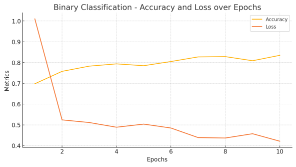

Figure 14: Loss and Accuracy at each training epoch for CNN binary classifier

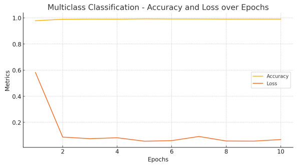

Figure 15: Loss and Accuracy at each training epoch for CNN multiclass classifier

4 Conclusion

This study explored the application of machine learning models for breast cancer diagnosis using two major data sets: the Wisconsin Breast Cancer Diagnosis data set and Breast Cancer Histopathological Database (BreakHis). After reviewing the literature on machine learning models applied to the Wisconsin Breast Cancer Diagnosis dataset, we decided to implement decision trees with bagging. To run our model on the BreakHis dataset, we implemented Otsu's thresholding method to extract data from the images. We then ran our model on the feature-extracted data and compared its performance with a CNN model applied to the raw image dataset. While the CNN model exhibited superior performance, they require significant computational resources and large datasets, which may limit their immediate adoption in some clinical settings. We, therefore, hope to use the CNN results as a baseline for improving our feature extraction and our model. The model we used proved to be an effective method for structured data, achieving competitive accuracy while being computationally efficient. However, its reliance on feature extraction highlights the importance of improving preprocessing steps when applying traditional machine learning techniques.

Overall, this work reinforces the importance of selecting appropriate machine learning models based on the specific requirements of accuracy, interpretability, and computational feasibility. While CNNs are currently a more accurate method for breast cancer diagnosis, decision trees could prove to be a practical alternative in resource-constrained environments. The promising accuracy achieved by combining traditional machine learning algorithms with feature extraction demonstrate that this approach can be a strong alternative to CNNs for image datasets.

4.1 Future Work

This study has demonstrated the potential of both decision trees and CNN models for breast cancer diagnosis, and there are several avenues for future research to further enhance performance and applicability.

The reliance of decision trees on feature extraction emphasizes the need for more robust and automated preprocessing methods. Exploring advanced image processing techniques, such as edge detection, texture analysis, or deep feature extraction, could improve the quality and relevance of input features, thereby enhancing the algorithm's accuracy and consistency. With more time, we could delve deeper into Otsu thresholding and other feature extraction methods and compare the model outputs given various feature sets.

Another constraint in our current work is that the effectiveness of CNNs is closely tied to the availability of large datasets. To address the challenge of limited labeled data in the BreakHis dataset, future work could focus on implementing data augmentation techniques or leveraging pre-trained models through transfer learning. These approaches can improve CNN performance while reducing the dependency on extensive computational resources.

Finally, machine learning models, especially CNNs, often lack interpretability, which is crucial in clinical decision-making. Future research could incorporate explainability techniques, such as SHAP (SHapley Additive exPlanations) or Grad-CAM (Gradient-weighted Class Activation Mapping), to provide more transparent insights into model predictions.

By addressing these areas, future research can build on the findings of this study, driving the development of more effective, accessible, and interpretable machine-learning solutions for breast cancer diagnosis.

References

[1] Md Toukir Ahmed, Md Niaz Imtiaz, and Animesh Karmakar. 2020. Analysis of Wisconsin Breast Cancer original dataset using data mining and machine learning algorithms for breast cancer prediction. *Journal of Science Technology and Environment Informatics* 9, 2 (2020), 665–672.

[2] Laith Alzubaidi et al. 2021. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Computers in Biology and Medicine* 131 (2021), 104035.

[3] Ahmad Taher Azar and Shereen M El-Metwally. 2013. Decision tree classifiers for automated medical diagnosis. *Neural Computing and Applications* 23 (2013), 2387–2403.

[4] Lucas Rodrigues Borges. 2015. Analysis of the Wisconsin breast cancer dataset and machine learning for breast cancer detection. *Group* 1, 369 (2015), 15–19.

[5] M. Divyavani and G. Kalpana. 2021. An Analysis on SVM & ANN Using Breast Cancer Dataset. *Aegaeum Journal* 8, 12 (2021), 369–370. https://www.researchgate.net/publication/348869189

[6] Kaggle. 2016. Breast Cancer Wisconsin (Diagnostic) Data Set. https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data.

[7] Kaggle. 2019. BreakHis Dataset. https://www.kaggle.com/datasets/ambarish/breakhis.

[8] Disha Mehta, Aakash Mohite, Vaishnavi Shinde, Ritika Khatri, and Indu Dokare. 2022. Detection of Breast Cancer using Machine Learning Algorithms. *Available at SSRN 4108758* (2022).

[9] National Breast Cancer Foundation (NBCF). 2024. Breast Cancer Facts & Stats. https://www.nationalbreastcancer.org/breast-cancer-facts/.

[10] World Health Organization. 2024. Breast cancer-WHO Fact Sheet. https://www.who.int/news-room/fact-sheets/detail/breast-cancer.

[11] American Cancer Society. 2021. Breast Cancer Facts & Figures 2021-2022. (2021). https://www.cancer.org/research/cancer-facts-statistics/breast-cancer-facts-figures.html

[12] Y. Y. Song and Y. Lu. 2015. Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry* 27(2) (2015), 130–135. https://doi.org/10.11919/j.issn.1002-0829.215044

[13] Utkarsh Prakash Srivastava, Vidushi Vaidehi, Tawal Kumar Koirala, and Palash Ghosal. 2023. Performance Analysis of an ANN-based model for Breast Cancer Classification using Wisconsin Dataset. In *2023 International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC)*. 1–5. https://doi.org/10.1109/ISACC56298.2023.10083642

[14] Fadi Thabtah, Suhel Hammoud, Firuz Kamalov, and Amanda Gonsalves. 2020. Data imbalance in classification: Experimental evaluation. *Information Sciences* 513 (2020), 429–441.

[15] Muhammad Umer, Mahum Naveed, Fadwa Alrowais, Abid Ishaq, Abdullah Al Hejaili, Shtwai Alsubai, Ala'Abdulmajid Eshmawi, Abdullah Mohamed, and Imran Ashraf. 2022. Breast cancer detection using convoluted features and ensemble machine learning algorithm. *Cancers* 14, 23 (2022), 6015.

[16] Xiangyang Xu, Shengzhou Xu, Lianghai Jin, and Enmin Song. 2011. Characteristic analysis of Otsu threshold and its applications. *Pattern recognition letters* 32, 7 (2011), 956–961.

[17] J. Zou and Y. Han. 2008. Overview of Artificial Neural Networks. *Livingstone, D.J. (eds) Artificial Neural Networks. Methods in Molecular Biology™* 458 (2008). https://doi.org/10.1007/978-1-60327-101-1_2

Statement of Work

4.1.1 *Danny Smyth.*

- Helped organize group meetings
- Algorithm analysis and comparison of decision trees and ANN
- Helped interpret, analyze, and write up results
- Model construction: helped with model formulation including the introduction and implementation of bagging to decision trees
- Presentation: Discussed algorithm analysis, Breakhis data set, and project goals.
- Helped with paper write-up

4.1.2 *Kimiya MohammadiJozai.*

- Helped organize meetings and deliverables
- Wrote the code
- Visualized the results
- Helped with the presentation
- Designed the website
- Helped with paper write-up

4.1.3 *Umur Kose.*

- Helped with the website
- Helped with presentation
- Helped with paper write-up

4.1.4 *Vanessa Eichensehr.*

- Literature Review of ML Algorithms implemented on WBCD
- Algorithm Analysis for Decision Tree and ANN
- Supported interpretation and analysis of results
- Discussed introduction, related works, and methodology in presentation
- Helped with website
- Helped with paper write-up