Breast Cancer Detection Using Machine Learning Techniques

1Anju Kumari, 2Mohana Vani S, 3Namitha V Pawar, 4Vijay Kumar S.

Assistant Professor
1Information Science and Engineering Department,
2BNM Institute of Technology, Bangalore, India

Abstract—Breast cancer is the second leading cause of demise for women, so correct early detection can assist lower breast most cancers mortality prices. This paper targets at giving an overview of breast most cancers detection using machine learning techniques. The methodology used here is a Convolutional Neural network with a pre-trained model. For the implementation of the ML algorithms, the dataset was partitioned into the schooling section and the checking out section.

Index Terms—System Design and Development, Machine Learning, Feature Extraction.

I. INTRODUCTION
Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes. Machine learning algorithms use historical data as input to predict new output values. The Internet of things describes physical objects that are embedded with sensors. It enables various applications such as Breast Cancer Detection Using Machine Learning Techniques. Cancer is a creation of abnormal cells that come from a modification in these cells genetically and spreads into the body, a late in diagnosis and treatment leads to death. There are two types of breast cancer, invasive and non-invasive. The former is harmful, malignant, ability to infect other organs, and classified as cancerous.

II. LITERATURE SURVEY
Breast Cancer can be detected using various methods and algorithms. In the paper [1], Deep Neural Network with Support Value (DNNS) is introduced to produce better quality images and to fix other performance parameters. The authors propose a new algorithm or pseudocode along with mathematical formulas to evaluate the efficiency and performance. Unlike other methods, the proposed method is based on Support value on a deep neural network. To meet the better performance, efficiency, and quality of images, a normalization process has been employed. Experimental results proved that the proposed DNNS is quite better than the existing methods. It is ensured that the proposed algorithm is advantageous in both performance, efficiency, and quality of images are crucial in the latest medical systems. Deep Neural networks usually require much more data than traditional machine learning algorithms. Deep neural networks are also more computationally expensive than traditional algorithms in machine learning. There is no standard theory to guide you in selecting right deep learning tools as it requires knowledge of topology, training method and other parameters. In this paper [2], Proposed methodology in paper is ML methods used were: Artificial Neural Network (ANN), Extreme Learning Machine (ELM), Support Vector Machine (SVM) and K-Nearest Neighbors (k-NN). In addition, the hyperparameter values giving the least errors for ANN, ELM, k-NN and SVM methods are determined using Hyperparameter optimization technique. The importance of this work is pretty high because of the usage of the different type of data. In addition, this study is also important because four different ML methods are compared. In addition, this study may support the further work in this field. The obtained accuracy rate cannot be regarded as very high. The k-NN method does not actually contain the training phase. [3], Proposed methodology in this paper is seven phases of Machine Learning are used in this paper. They are: Pre-Processing Data, Data preparation, Features Selection, Feature Projection, Feature Scaling, Model Selection and Prediction. Methods used are: Logistic Regression, K Nearest Neighbor (k-NN), Support Vector machine, the highest value of correctly classified instances and the lowest value of incorrectly classified instances than the other classifiers. K-NN is a lazy learner and does not do much during training process unlike other classifiers that build the models. Future enhancements in this paper is that further research in this field should be carried out for the better performance of the classification techniques so that it can predict on more variables. It is needed to reduce the error rates with maximum accuracy. [4], Proposed methodology in paper is Data Mining and Machine Learning Algorithms. They are: Decision tree algorithms, K-nearest-neighbours (KNN) algorithm, Support Vector Machine (SVM), Naive Bayes (NB) It is a probabilistic classifier and Logistic regression. UCIMachine Learning Repository for breast cancer dataset is used in this project. It is observed that a good dataset provides better accuracy. Data mining and machine learning can be a very good help in deciding the line of treatment to be followed by extracting knowledge from such suitable databases. K-NN is a lazy learner and does not do much during training process unlike others classifiers that build the models. Future enhancement of this paper is Selection of appropriate algorithms with good home dataset will lead to the development of prediction systems. [5], Proposed methodology in paper is the Search strategy, Inclusion and exclusion criteria, Data extraction, prediction model risk of bias assessment tool (PROBAST) this is the first system to be a view of the application of ML to breast cancer survival prediction, and accurate 5-year survival predictions are very important for further research. The models still face limitations related to a lack of data preprocessing steps, the excessive differences of sample feature selection, and issues related to validation and promotion. Future enhancement of this paper is the model performance still needs to be further optimized, other barriers should be addressed. Researchers and medical workers should connect with reality, choose a model carefully, use the model in clinical practice after verification, and use rigorous design and validation methods with a largesampleofhigh-qualityresearchdataonthebasisofpreviousfindings.

III. DATASET USED
The dataset which is used in this model is the Breast Cancer Histopathological Database (BreakHis) dataset. The Breast Cancer
Histopathological Image Classification (BreakHis) is composed of 9,109 microscopic images of breast tumor tissue collected from 82 patients using different magnifying factors (40X, 100X, 200X, and 400X). To date, it contains 2,480 benign and 5,429 malignant samples (700x460 pixels, 3-channel RGB, 8-bit depth in each channel, PNG format). This database has been built in collaboration with the P&D Laboratory – Pathological Anatomy and Cytopathology, Parana, Brazil.

IV. SYSTEM DESIGN AND DEVELOPMENT

Architectural Design: Requirements of the software should be transformed into an architectural that describes the software’s stop-level structure and identifies its components. This is accomplished through architectural design (also called system design), which acts as a preliminary blueprint from which software can be developed. IEEE architectural design is the process of defining a collection of hardware and software components and their interface to establish the framework for the development of a computing system. This framework is established by examining the software requirement document and designing a model for providing implementation details. These details are used to specify the components of the system along with their inputs, outputs, functions, and the interaction between them. An architectural design performs several functions.

V. IMPLEMENTATION

List of Modules
The project has been implemented by dividing the entire project into three modules. They are:

- Training
- Utilities
- Testing

Training: The training file follows an abstraction model of programming. This file systematically calls all the functions required for the project to execute successfully and create a model. Following a modular approach, this file has many steps and function calls as listed below:

- Importing data
- Data visualization and distribution
- Preparing data for processing
- Splitting data
- Adding more variety and variance to the data
- Preprocessing
- Creating a batch generator for training
- Creating the model
- Saving the model to disk
- Model analytics

Algorithm: This section explains in detail about the concept of convolutional neural networks and later the proposed neural network model used in the project and that is talked about in this section, along with important libraries used throughout the project.

Convolutional Neural Network (CNN): comprises one or more convolutional layers. It is proceeded by one or more neural network connected layers. The purpose of this layer is to receive a feature map. Usually, we start with a low number of filters for low-level feature detection. The deeper we go into the CNN, the more filters we use to detect high-level features. Feature detection is based on ‘scanning’ the input with the filter of a given size and applying matrix computations in order to derive a feature map.
Pooling is a method of arbitrary experimenting and is commonly used while adding pooling layers to lower the parameters and to eliminate unnecessary features during the training in a pursuance to avoid overfitting in the network. After the convolutional layers, the multiple proportional arrays are flattened into a two-dimensional array in a fully connected network.

**Pooling Layer:** Similar to the Convolutional layer, the pooling layer is responsible for reducing the spatial size of the convolved feature. This is to decrease the computational power required to process the data through dimensionality reduction. Furthermore, it is useful for extracting dominant features which are rotational and positional invariant, thus maintaining the process of effective training of the model.

**Fully Connected Layer:** In a fully connected layer, we flatten the output of the last convolution layer and connect every node of the current layer with the other nodes of the next layer. Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular Neural Networks and work in a similar way. Adding a fully connected layer is a cheap way of learning non-linear combinations of the high-level features which can be represented by the output of the convolutional layer. The fully connected layer is learning a possibly non-linear by flattening the image into a column vector.

**TensorFlow:** TensorFlow is a license-free software for dataflow graphs to build models. It allows neural networks to go hand-in-hand with multi-layers and is predominantly used for prediction and classification of the sample data fed into the model or the network. TensorFlow provides excellent functionalities and services and enables the high-level complex parallel computation for building advanced neural networks.

**Keras:** It is also another python library majorly used in neural networks. It is more user-friendly, modular and enables faster experimentation with neural networks. It is can be made to run on top of TensorFlow. It contains a great number of neural network building blocks such as activation functions, optimizers, and tools compatible with working of text and image data.

**Flask:** It is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework-related tools.

**Utilities:**

This Python file contains the main logic of the entire project. Every task is executed in a different function, making the code modular and also attaining a good level of abstraction. All the functions calls are made in the training file of the project, as seen in section

**Testing:**

This Python script uses the already created and saved model. A server socket is created which listens to the appropriate port. The code in this file is responsible for taking in pictures from the simulator, preprocess it, pass it through the neural network’s model and send the generated acceleration and steering angle values back to the simulator’s agent.

VI. CONCLUSION AND SCOPE FOR FUTURE ENHANCEMENT

The diagnosis of breast cancer in an early stage can help in the reduction of the mortality caused by breast cancer. In this project, we have demonstrated how to classify benign and malignant breast cancer from a collection of microscopic images using convolutional neural networks. We implemented pre-trained CNN models with fine-tuning leveraging transfer learning to observe the classification performance of breast cancer from microscopic images. We evaluated the fine-tuned pre-trained models by applying ResNet50 with the Adam optimizers. Although this project is far from complete but it is remarkable to see the success of deep learning in such varied real-world problems. We have also proposed several strategies for training the CNN architecture, based on the extraction of patches obtained randomly by a sliding window mechanism, that allow to deal with the high-resolution of the textured images without changing the CNN architecture designed for low-resolution images. Our experimental results obtained on the Breast-Ki datasets showed improved accuracy obtained by the CNN when compared to traditional machine learning model trained on the same dataset but with state of the art texture descriptors.
Future work can explore different CNN architectures and the optimization of the hyperparameters. Also, strategies to select representative patches in order to improve the accuracy can be explored and also to validate the model with other datasets that include new images.

REFERENCES


