

Article

Dense Model for Automatic Image Description Generation with Game Theoretic Optimization

Sreela S R *  and Sumam Mary Idicula

Department of Computer Science, Cochin University of Science and Technology, Kochi, Kerala 682022, India; sumam@cusat.ac.in

* Correspondence: sreela148@cusat.ac.in; Tel.: +91-949-770-8518

Received: 11 October 2019; Accepted: 13 November 2019; Published: 15 November 2019



Abstract: Due to the rapid growth of deep learning technologies, automatic image description generation is an interesting problem in computer vision and natural language generation. It helps to improve access to photo collections on social media and gives guidance for visually impaired people. Currently, deep neural networks play a vital role in computer vision and natural language processing tasks. The main objective of the work is to generate the grammatically correct description of the image using the semantics of the trained captions. An encoder-decoder framework using the deep neural system is used to implement an image description generation task. The encoder is an image parsing module, and the decoder is a surface realization module. The framework uses Densely connected convolutional neural networks (Densenet) for image encoding and Bidirectional Long Short Term Memory (BLSTM) for language modeling, and the outputs are given to bidirectional LSTM in the caption generator, which is trained to optimize the log-likelihood of the target description of the image. Most of the existing image captioning works use RNN and LSTM for language modeling. RNNs are computationally expensive with limited memory. LSTM checks the inputs in one direction. BLSTM is used in practice, which avoids the problem of RNN and LSTM. In this work, the selection of the best combination of words in caption generation is made using beam search and game theoretic search. The results show the game theoretic search outperforms beam search. The model was evaluated with the standard benchmark dataset Flickr8k. The Bilingual Evaluation Understudy (BLEU) score is taken as the evaluation measure of the system. A new evaluation measure called GCorrect was used to check the grammatical correctness of the description. The performance of the proposed model achieves greater improvements over previous methods on the Flickr8k dataset. The proposed model produces grammatically correct sentences for images with a GCorrect of 0.040625 and a BLEU score of 69.96%

Keywords: image captioning; image description generation; deep learning; Densenet; bidirectional LSTM

1. Introduction

The World Wide Web is a data store with a vast collection of images. Image searching is a challenging task. Currently, most of the search engines use meta-data to search for images. The metadata of images includes user annotated tags, surrounding text, captions, etc. Not every picture on the Internet has metadata. Therefore, the automatic generation of image descriptions reduces the complexity of image searching. The applications range from supporting visually impaired users to human–robot interaction.

The depiction of an image contains scenes, specific objects, spatial relationships, and attributes for adding additional information about objects. The output of computer vision combined with language models is suitable for the image description generation process. Natural language Generation (NLG)

is one of the basic research issues in Natural Language Processing (NLP). It has a broad range of applications in machine translation, discourse, summarization, and machine assisted vision. Regardless of the significant progress in the most recent decade, NLG remains an open research issue. Previous sentence generation was done by either utilizing an n-gram language model or template based approach. As of now, RNN (Recurrent Neural Network) [1] and LSTM [2] are connected for NLG.

The essential steps of automatic image description generation are image parsing and surface realization. Image parsing is the process of generating features of the image. Surface realization is the process of generating an image description. The task of surface realization is to create relevant, understandable, and question specific content reports. Image parsing is done using a deep convolutional neural network. Deep recurrent neural systems are utilized for surface realization. The optimization, such as word selection, is done using two methods, game theoretic search and beam search. To describe an image effectively, all entities, attributes, and relationships are identified in the sentence and mapped to the complex visual scene.

Google's Neural Image Caption generator (Google NIC) [3] was used as the seminal paper for writing this paper. The NIC model used an encoder-decoder framework in which the encoder was InceptionV3 and the decoder was the recurrent neural network. Another important work was Andrej Karpathy's work [4] from Stanford University. The objective of the work was to generate a human readable image description using past and future contexts of trained captions. The proposed model differed from existing models in that it learned the semantics of the sentences using BLSTM's bidirectional nature and mapped sentence features to complex image features.

To achieve this objective, a framework for the automatic generation of image descriptions with two major components was proposed. They are explained as follows:

- **Image parsing:**
Image parsing is done using Densenet. The advantages of using Densenet are as follows: elimination of the vanishing gradient problem, reinforcing feature propagation, supporting feature reuse, and reducing the number of parameters.
- **Surface realization:**
Two BLSTMs are used to implement surface realization. First, BLSTM is implemented for language modeling and the other for caption generation. The significance of BLSTM is that it investigates the past and future reliance to give a forecast. Earlier, the word selection in caption generation was performed using beam search. In our work, beam search and game theoretic search are implemented, and it is found that game theoretic search outperforms beam search.

A grammatical accuracy measure called GCorrectis used to evaluate the grammatical correctness of the generated description.

The paper is structured as follows. A review of the existing works is discussed in Section 2. Section 3 explains the mathematical foundation of the proposed model. The implementation and experiment results are explained in Sections 5 and 6. The paper is concluded in Section 7.

2. Related Work

Automatic image description generation is a core part of image understanding. Several approaches have been developed for this task. These systems are categorized based on the generation methods and image feature extraction methods. Different kinds of image captioning systems are discussed in Section 2.1. The image captioning systems became more advanced after the deep learning models become popular for object detection. The deep neural networks are a powerful tool for image feature extraction and are explained in Section 2.2.

2.1. Image Captioning

Image description generation models are categorized into three types. These are direct generation models, visual space retrieval models, and multimodal space retrieval models [5]. The general approach

of the direct generation model is to predict the semantics of the image using the visual features first and then to generate sentences corresponding to these semantics. Midge [6] and BabyTalk [7] are based on direct generation models. In the visual space retrieval model, the portrayal is created by retrieving similar images and transferring captions from the visual space to a new image. The Im2Textmodel [8] uses this approach. The multimodal space retrieval model treats the description problem as a ranking problem. The methods of Hodosh et al. [9], Socher et al. [10], and Karpathy et al. [4] followed this approach. Some image captioning systems use a visual attention mechanism [11].

Image captioning systems can be classified into two types based on the feature extraction methods. They are layout based approaches and deep neural network based approaches.

2.1.1. Layout Based Approaches

In this approach, the captions are generated using the outputs of object detection, attributes, and scene identification. Farhadi et al. [12] used the Markov random field, GIST, and Support Vector Machine (SVM) for caption generation and converted the scene elements to text using layout. Kulkarni et al. [7] utilized the Conditional Random Field (CRF) to associate the objects, attributes, and prepositions. Midge [6] utilized the Berkeley parser and Wordnet ontologies to generate text. The disadvantage of these systems is the generation of incorrect captions due to inaccurate object detection. These systems used traditional machine learning algorithms for object detection, which resulted in poor performance.

2.1.2. Deep Neural Network Based Approaches

The image captioning system involves image to text translation. Currently, most of the image captioning systems adopt the encoder-decoder framework. The image feature extraction is done in the encoding phase. The encoding is done using deep neural networks, which give better performance in object detection. The decoder is implemented using recurrent neural networks or LSTM, which is used for caption generation. Kiros et al. [13] used feed-forward neural networks and multimodal log-bilinear models to generate words from previous words. Some systems utilized the recurrent neural network for caption generation. Gong et al. [14] used deep CNN and bag of words for description generation. Karpathy [15] utilized Region level CNN (RCNN) and bidirectional RNN for whole description generation. Vinyal et al. [3] used LSTM as a decoder. Donnelly et al. [16] produced a multimodal architecture for caption generation. Sou et al. [17] provided linguistic importance to the interaction between learned word embeddings and the LSTM hidden states. Wang et al. [18] built a deep Convolutional Neural Network (CNN) and two separate LSTM networks for caption generation. You et al. [19] used top-down and bottom-up approaches for image captioning.

Top-down visual attention mechanisms are commonly used in image captioning. Peter Anderson's method [20] was based on a hybrid of the top-down and bottom-up visual attention mechanism. Agashi Poghosyan [21] modified the LSTM cell to get greater significance of image features. The convolutional image captioning [22] system uses convolutional networks for captioning. The Groupcap system [23] considers the diversity of image captions. Phi-LSTM [24] predicts image captions from phrase to sentence. Han's system [25] used a fast image captioning system using YOLO and LSTM. Text based visual attention has also been implemented in image captioning systems [26].

2.2. Deep Neural Network

Currently, deep neural networks have an important role in caption generation. Deep convolutional neural networks [27] are feed-forward neural networks with a convolution operation instead of multiplication. The disadvantage of deep CNN is the difficulty in training due to the vanishing gradient problem. The variants of deep CNN are LeNet [28], AlexNet [29], VGGNet [30], GoogLeNet [31], Residual Neural Network (ResNet) [32], highway network [33], Densenet [34], MobileNet [35], SqueezeNet [36], etc. VGGNet was developed and trained by the Visual Geometry Group (VGG)

at the University of Oxford. ResNets have shortcut connections parallel to convolutional layers. The gradients are backpropagated through shortcut connections. Therefore, ResNet has no vanishing gradient problem, and thus, the training is faster. Recurrent Neural Networks (RNN) [1] and long short-term memory are used for sentence generation. RNNs are good for short contexts because they manage short term dependencies and have the vanishing gradient problem. Long haul expectations are hard for RNN. Therefore, LSTMs come into the picture. They consider long term dependencies for sequence prediction. A word embedding is a learned description of text where similar kinds of words have the same representation. It is also a technique for representing words in a vector form using a predefined vector space. It also shows the progress of deep learning in solving challenging natural language problems. The examples of word embedding models are bag of words, TF-IDF, distributed embeddings, embedding layer, Word2vec [37], GloVe [38], and FastText [39]. Bag of words, TF-IDF, and distributed embeddings are traditional word embedding models. Word2Vec, GloVe, and FastText are neural network based word embeddings.

2.3. Game Theory

Game theory [40] is a branch of mathematics that models the conflicts and coordination between different players in a scenario. In game theory, a problem is formulated as a game. The game can be cooperative and non-cooperative. In the non-cooperative game, individual players are the decision-makers, and so, they do not make alliances with other players in the game. The critical concept in non-cooperative game theory is the Nash equilibrium, which was introduced by John Forbes Nash. Cooperative games are games in which coalitions between players result in profits in the game.

2.3.1. Cooperative Game Theory

Cooperative games are games in which players should form alliances due to cooperative behavior. The main difference between cooperative and non-cooperative games is that coalitions are formed between players. These games are mathematically defined with a value function $v : 2^N \rightarrow R$, where N is the number of players. The measures such as the Shapley value and core value determine the payoffs for players in each coalition [41]. Sun [41] developed a feature selection method using cooperative game theory. In this work, cooperative game theory is used.

3. System Architecture

Let I be the image and D be the description that is represented as $\langle D_1, D_2, D_3, \dots, D_n \rangle$, and the proposed model optimizes the log-likelihood of description D given image I . The parameters of the model can be expressed as $\theta^* = \log P(D_n | I; D_1, D_2, \dots, D_{n-1})$. In this model, for each iteration, the probabilities of each word in the vocabulary is calculated.

The architecture of the proposed model is depicted in Figure 1. The proposed model is a combination of the image model, language model, and caption model, which are explained in Sections 3.1–3.3. The inputs to the model are the image and a partial caption. The image is a numerical array of size $224 \times 224 \times 3$, and the partial caption is a numerical vector of length 40. The image model encodes the image as features of size 40×128 . The language model converts the image description into an encoded vector of sentence features. CNN in the image model encodes the image into features, and BLSTM in the caption model produces the next word from the image features and sentence features. The caption model initially computes the log probability of the words in the vocabulary, selects the word using game theoretic search and beam search, and generates the description of the image.

3.1. Image Model

The image model extracts the features of the image and reduces the size of the image features. The important components of the image model are Densenet [34], the dense layer, the activation layer, and the repeat layer. Figure 2 explains the architecture of the image model.

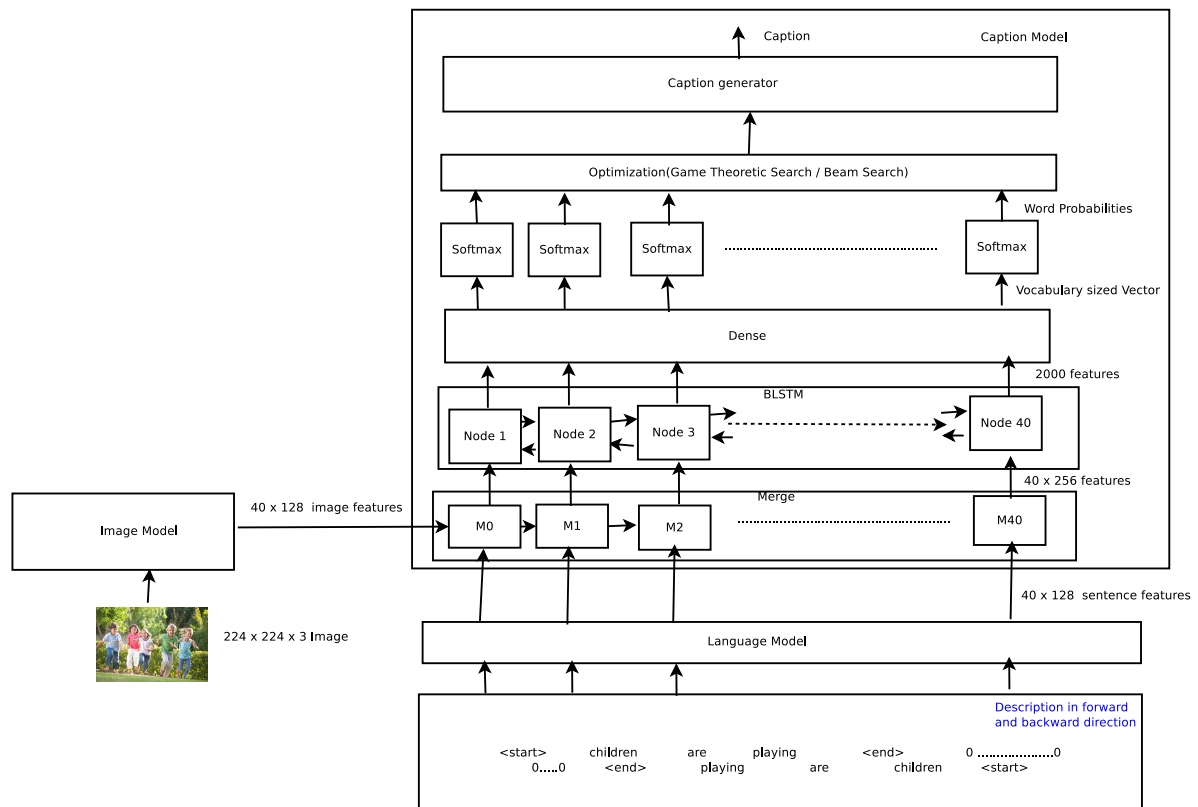


Figure 1. Architecture of the proposed model.

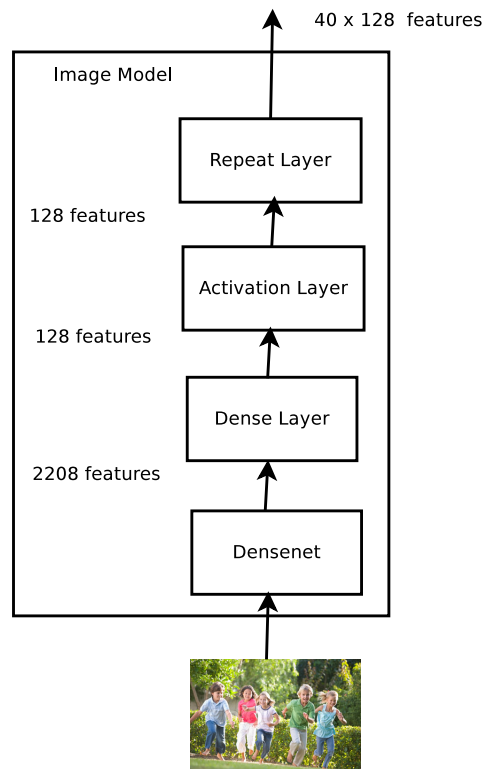


Figure 2. Architecture of the image model.

3.1.1. Densenet

Densenet was used to extract the visual content of the image. It is a type of deep convolutional neural network. In Densenet, the input of the l^{th} layer is the output of all preceding layers and is expressed by Equation (4). It provides an excellent object classification rate. The architecture of Densenet is shown in Figure 3.

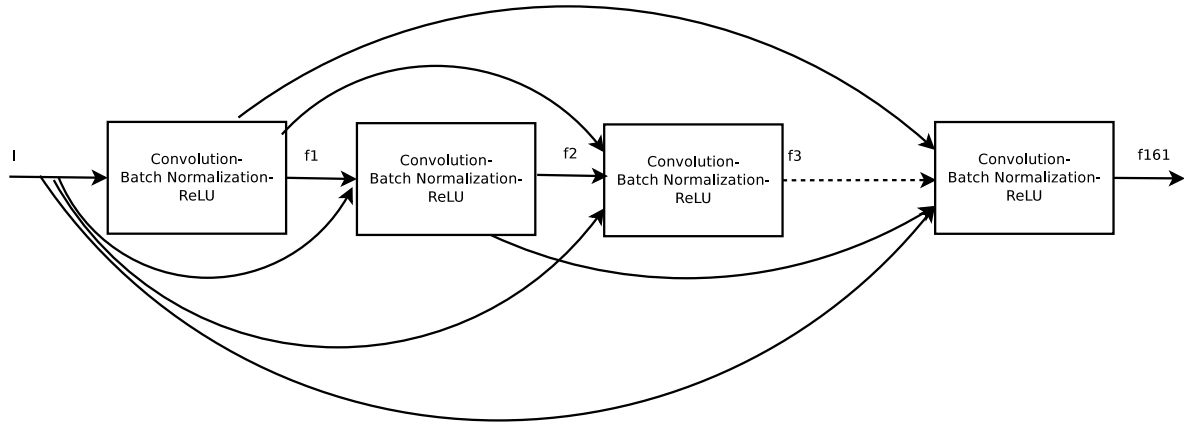


Figure 3. Architecture of Densenet.

Equation (4) defines the operation of Densenet. The function D is a combination of convolution, batch normalization, and ReLU functions. Convolution is an element-wise multiplication of image and kernel matrices. Batch normalization is the operation of shifting inputs with zero mean and unit variance in each mini-batch. After that, the elements with negative values are converted to zero using the ReLU function.

$$Densenet(I) = D_l([I, f_1, f_2 \dots, f_{l-1}]) \quad (1)$$

where I is the image and f_1, f_2, f_{l-1} are the features of the first, second, and $l - 1$ th layers, respectively.

Densenet was used as the image feature encoder to get the spatial representation of images. In the experiments, the image model used Densenet with 161 layers. The input of Densenet had dimensions of $224 \times 224 \times 3$, which represent the dimensions of the image. The output of the last layer of Densenet had a dimension of 2208 in this work.

3.1.2. Dense Layer

This was a fully connected neural network layer for feature reduction. The input of this layer was features extracted from Densenet. This layer reduced the feature size from 2208 to 128. The output of the dense layer (x_{-1}) is defined by Equation (2).

$$x_{-1} = W_d \cdot Densenet(I) \quad (2)$$

where W_d is a kernel weights matrix having dimensions of 128×2208 .

3.1.3. Activation Layer

The output of the dense layer was applied to the activation layer to get a fixed size positive vector using the ReLU function. The ReLU function replaced negative values with zeros in the input. Equation (3) defines the functionality of this layer.

$$ReLU(x) = \max(0, x) \quad (3)$$

3.1.4. Repeat Layer

The repeat layer was used for repeating the data up to the maximum caption length for merging purposes, which is represented by Equation (4).

$$x_0 = \text{repeat}(x_{-1}, \text{max_cap_len}) \quad (4)$$

where max_cap_len is the maximum caption length.

3.2. Language Model

The caption was encoded as a vector of language features in the language model phase. The language model was implemented using the embed layer, Bidirectional LSTM (BLSTM) [18], and the time distributed dense layer. The architecture of the language model is shown in Figure 4.

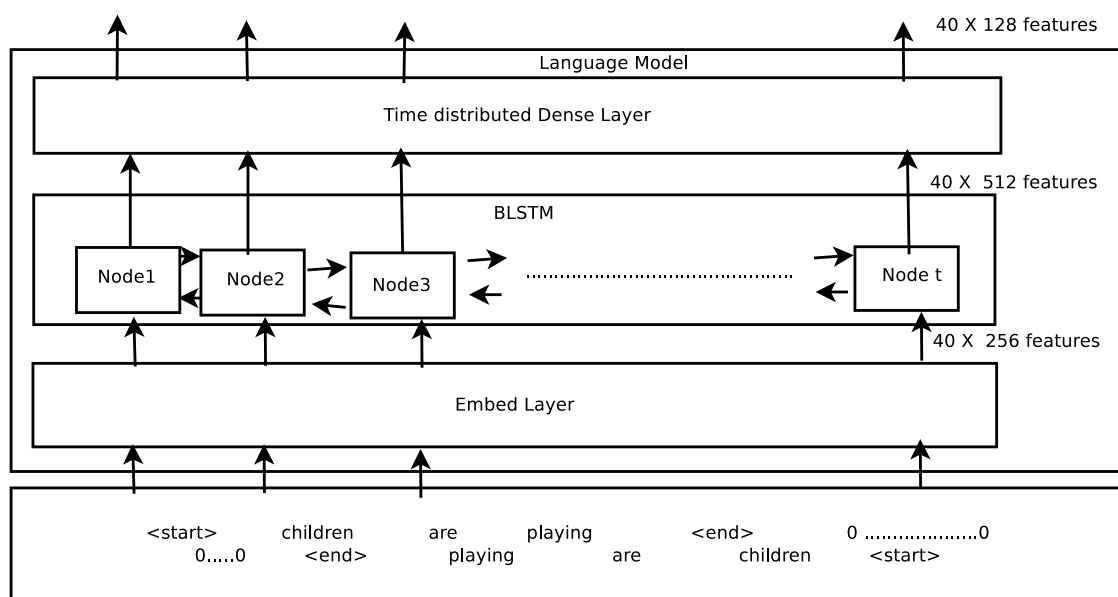


Figure 4. Architecture of the language model.

3.2.1. Embed Layer

The captions were preprocessed and given to the embed layer. The functionality of the embed layer was to represent each word using a word embedding of size $\text{max_cap_len} \times S_i$ where S_i is the fixed size of LSTM input and max_cap_len is the maximum caption length. In the experiment, the value of S_i was 256. The output of the embed layer was given to the BLSTM.

3.2.2. Bidirectional LSTM

Long Short Term Memory (LSTM) [2] is a type of recurrent neural network architecture that eliminates the vanishing gradient problem in RNN and allows for learning long term sequences. The memory blocks are in charge of recalling things, and control of this memory is done through three multiplicative units called gates. The gates are the input, forget, and output gates. The input gate is responsible for adding information to the cell state. The forget gate expels data from a cell state. The output gate functions to choose useful information from the current cell state as an outcome.

The BLSTM looks in the forward and backward direction of the caption. By examining the two paths, it utilizes the past and future information for modeling the current frame. The language model is described by the equation:

$$h_{L_t} = \overleftrightarrow{\text{LSTM}}(c_t, h_{t-1}, m_{t-1}) \quad (5)$$

The BLSTM is a combination of forward and backward LSTMs. Therefore, \overleftarrow{LSTM} is defined by using forward and backward LSTM equations. Forward LSTM is defined by the following equations.

$$f_t = \sigma_g(k_f c_t + U_f h_{t-1} + b_f) \quad (6)$$

$$i_t = \sigma_g(k_i c_t + U_i h_{t-1} + b_i) \quad (7)$$

$$o_t = \sigma_g(k_o c_t + U_o h_{t-1} + b_o) \quad (8)$$

$$m_t = f_t \circ m_{t-1} + i_t \circ \sigma_m(k_m c_t + U_m h_{t-1} + b_m) \quad (9)$$

$$h_t = o_t \circ \sigma_h(m_t) \quad (10)$$

Backward LSTM is defined by the following equations.

$$f_t = \sigma_g(k_f c_t + U_f h_{t+1} + b_f) \quad (11)$$

$$i_t = \sigma_g(k_i c_t + U_i h_{t+1} + b_i) \quad (12)$$

$$o_t = \sigma_g(k_o c_t + U_o h_{t+1} + b_o) \quad (13)$$

$$m_t = f_t \circ m_{t+1} + i_t \circ \sigma_m(k_m c_t + U_m h_{t+1} + b_m) \quad (14)$$

$$h_t = o_t \circ \sigma_h(m_t) \quad (15)$$

where the initial values are $m_0 = 0$, $h_0 = 0$, and the operator \circ denotes the Hadamard product. c_t is the input vector to the LSTM unit; f_t : activation vector of forget gates; i_t : activation vector of input gates; o_t : activation vector of output gates; h_t is the output vector of the LSTM unit; m_t is the cell state vector; K , U , and b are weight matrices and the bias vector, respectively; σ_g is the sigmoid function; σ_m and σ_h are hyperbolic tangent functions.

In this work, the number of cells in BLSTM was the maximum caption length. The maximum caption length was set as 40. The output dimension of BLSTM was 40×512 .

3.2.3. Time Distributed Dense Layer

This is a time distributed fully connected layer, which means the operation is applied to every temporal slice of an input. This layer eases the burden of the network by limiting weights because one time step is prepared at once. This layer gives the sentence feature having a size of 40×128 .

3.3. Caption Model

The main components of the caption model are the merge layer, BLSTM, the dense layer, the softmax layer, beam search, and the caption generator. The outputs of the image and language models were merged using the row axis with the output having a size of 40×256 and given as the input to the caption model BLSTM.

Captions were modeled using bidirectional LSTM. The features of subtitles were fed to the model sequentially. The following equation defines BLSTM. It had 2000 features as the output.

The softmax function was used to calculate the score for each vocabulary item. The output of this layer was a vector containing the probability of each vocabulary item. It is described in Equation (16).

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad \text{for } j=1, \dots, K \quad (16)$$

Caption Generator

Game theoretic search and beam search are two algorithms implemented to find the right words for the description. The game theoretic search algorithm is described in Section 4. Beam search

was used to find the right words for the description and the best description. It is an optimization of the best-first search algorithm. It is commonly used in machine translation systems to find the best translation. In beam search, a predetermined number of best partial solutions are taken as candidate solutions, and the predetermined number is called the beam size (width). In the experiment, a beamwidth of three was used. The words with the top three scores were taken for beam search, and partial solutions were obtained based on these words. The probability scores of each partial solutions were computed by adding the previous probability scores of the partial solution and the probability of each resultant word. The partial solutions with the top three scores were taken as the three best partial captions. In this phase, the best caption was found by taking the caption with the maximum probability score from the beam search results. The best caption was taken as the output of the system. The main drawback of beam search is that it may lead to an optimal result or sometimes not find the solution. Game theoretic search overcomes these problems.

4. Game Theoretic Algorithm for Caption Generation

The game theoretic algorithm was used to extract the appropriate word for the caption of the image. Word selection was treated as a cooperative game. In word selection, partial captions were treated as players, and the probability value was a payoff value for each partial caption. A partial caption is a meaningful combination of words. The Shapley value is an essential measure for cooperative games. The game theoretic algorithm for word selection is explained in Algorithm 1. In this algorithm, the relationships between words are determined from the Shapley value of the coalition of words. The algorithm for the Shapley value calculation for cooperative games is explained in Algorithm 2. In Algorithm 1, the victory criterion is used to find the importance of a word over others in each iteration. The victory criterion was computed by the product of the normalized Shapley value and the value of the criterion. Information gain was treated as the criterion function. It is calculated using the Equation (17). The Shapley value determined the inherent impact of a player in the whole player set, which was used to control the relative importance of the value of the criterion.

$$C(i) = -\log(P_i) \quad (17)$$

where P_i is the probability score of the partial caption i .

Algorithm 1: Cooperative game theoretic algorithm for word selection or best caption selection.

Input: T (players (partial captions or captions), probability score ($P(1,2,3,\dots,n)$)), n is the number of players.

Output: selected partial captions or best caption

1. Calculate Shapley value for each player (ϕ).
 2. Initialize parameters $k = 0, j = 0, S = \{\}$.
 3. Randomly select σ value from the set $\{1, 2, 3, \dots, n\}$.
 4. Repeat from Steps 4–12 until j becomes σ .
 5. **for each player** i **in set of players** $N = \{1, 2, 3, \dots, n\}$. **do**
 6. Calculate the value of criterion $C(i)$ as information gain;
 7. Calculate its victory criterion $V(i) = C(i) \times \phi(i)$;
 - end**
 8. Choose the player i with the largest victory criterion.
 9. Remove i from the player set.
 10. $S = S \cup \{i\}$
 11. $j = j + 1$
-

In Algorithm 2, the Shapley value of each coalition is computed using Equation (18). $\Delta_i(K)$ is computed using Equation (19).

$$\phi_i(v) = \sum_{K \subset N} \Delta_i(K) \times \frac{\text{len}(K)!(n - \text{len}(K) - 1)!}{n!} \quad (18)$$

$$\Delta_i(K) = v(K \cup i) - v(K) \quad (19)$$

Algorithm 2: Shapley value calculation algorithm.

Input: set of players $N = \{1, 2, 3, \dots, n\}$

Output: Shapley values for each player ϕ

1. Initialize Shapley values for each player to zero.

2. **for** each player i in N **do**

 3. Create all subsets $\{\pi_1, \pi_2, \dots, \pi_t\}$ that contain players except i .

 4. **for** each subset π_j in $\{\pi_1, \pi_2, \dots, \pi_t\}$ **do**

 5. Calculate the value of $\Delta_i(\pi_j)$

end

 6. Calculate the Shapley value $\phi_i(v)$.

end

7. Normalize the vector ϕ .

The partial caption with the largest value of the victory criterion was taken as the final caption of the system. The victory criterion was obtained from the information gain and Shapley value. The game theoretic algorithm eliminated the problems of beam search, such as the sometimes infinite time needed for optimum results. This algorithm enhanced the image captioning system accuracy and Bilingual Evaluation Understudy (BLEU) score.

5. Implementation

The framework was implemented using Keras (<https://keras.io/>), Tensorflow (<https://www.tensorflow.org/>), and Python (<https://www.python.org/>). Keras is a high end deep learning library. The backend of Keras is Tensorflow, which is a package for dataflow programming and machine learning.

5.1. Training Details

The transfer learning mechanism was used to extract Densenet features of images by using pre-trained model weights from ImageNet. The language model used single-layer bidirectional LSTM with hidden size 256. Single-layer bidirectional LSTM with hidden size 1000 was used in the caption model. The model was trained at different epochs. The model achieved minimum validation loss at 50 epochs. Therefore, the model was fine-tuned with 50 epochs. In training, a random training data generator was used at each time due to computational resource constraints. The model was trained with an NVIDIA Tesla K80 GPU.

5.2. Optimization

The optimization function used was rmsprop, which is a commonly used optimization function in this type of work. rmsprop is an algorithm that divides the learning rate for weight by a running average of the magnitudes of new gradients for that weight.

6. Experiment

6.1. Datasets

Flickr8k [9] was used as the dataset for this work. It contains 8000 images, and each image has five captions. Out of the 8000 images, the training set included 6000 images, the validation set 1000 pictures, and remaining used for testing purposes.

6.2. Preprocessing

In the preprocessing stage, unnecessary words were removed using stop word removal. The vocabulary of the training set was created. The total size of the vocabulary was 8256. The dictionary of the vocabulary was defined using {word, index} mapping. The training set needed to be arranged for sequence learning. The training data had the form {image, partial caption} as x and next word as y. For example, if the caption is “⟨start⟩ He is playing ⟨end⟩”, ⟨start⟩ and ⟨end⟩ are used as starting and ending delimiters; partial captions are { ⟨start⟩, ⟨start⟩ He, ⟨start⟩ He is, ⟨start⟩ He is playing }; and next words are { He, is, playing, ⟨end⟩ }. The values of y were one hot encoded.

6.3. Performance Evaluation of the Model

The metrics for evaluating the model were accuracy and loss. The accuracies of different models are plotted in Figure 5. The X-axis shows the number of epochs, and the Y-axis shows the accuracy in percentage form. The accuracy grew increasingly. From the plots of accuracies, Densenet with BLSTM had an accuracy of 75.6%.

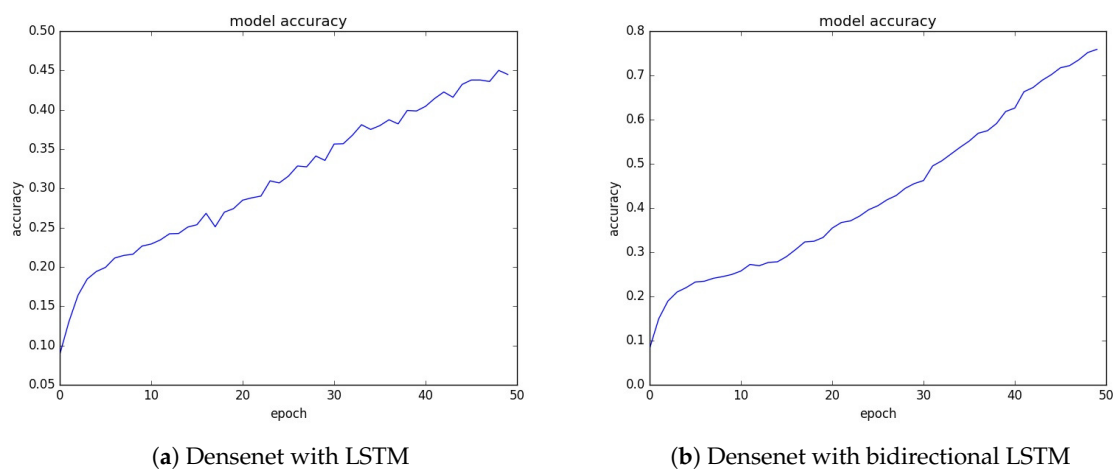


Figure 5. Comparison of the accuracies of different models.

The losses of different models are depicted in Figure 6. The loss was calculated using categorical cross-entropy. The loss decreased when the number of epochs increased. The loss of Densenet with BLSTM was 0.86. The parameters were well normalized throughout the training.

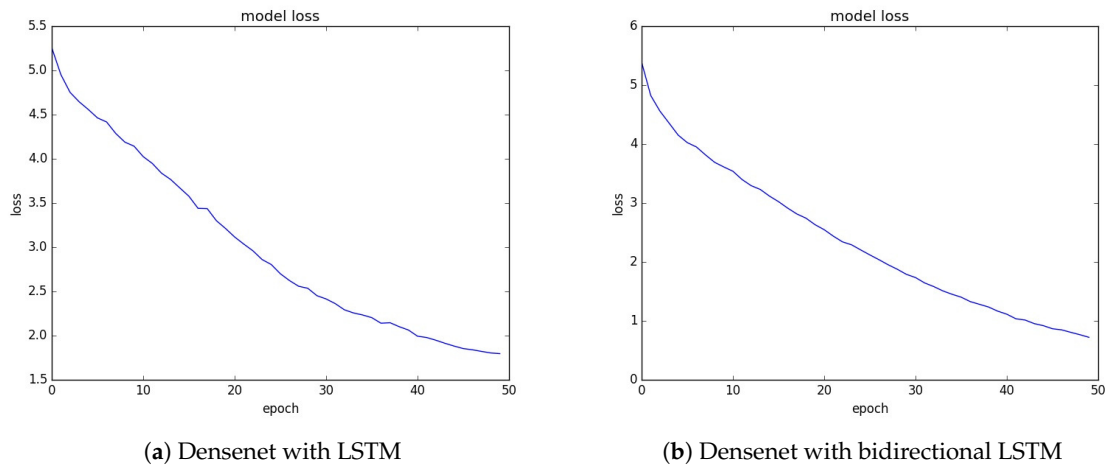


Figure 6. Comparison of the losses of different models.

From the experiment, Densenet with bidirectional LSTM gave better efficiency concerning accuracy and loss. The performance of the model was also evaluated using the BLEU score and GCorrect, which are discussed in Sections 6.4 and 6.5.

6.4. Generated Captions

The generated description was evaluated using the BLEU [42] score. The BLEU score is a machine translation evaluation measure that evaluates the generated description with the n-grams of the human description of images. It is computed using the Natural Language Toolkit (NLTK) (<http://www.nltk.org/>) package. The generated captions were divided into three kinds, successful, partially successful, and unsuccessful, based on the BLEU scores. The classification was based on Table 1. The thresholds for the BLEU scores were set manually.

Table 1. Classification of generated captions based on the Bilingual Evaluation Understudy (BLEU) scores.

BLEU-1	BLEU-2	BLEU-3	BLEU-4	Class
>0.8	>0.6	>0.5	>0.4	Successful
≤0.7	≤0.5	≤0.4	≤0.3	Unsuccessful
	Otherwise			Partially successful

The test image sample results are shown in Tables 2–4. The results were divided into three categories, successful image results, partially successful, and unsuccessful results. The successful image results are shown in Table 2. Partially successful results are depicted in Table 3. The unsuccessful results are shown in Table 4.

Table 2. Successful results.




Sl. No	Image	Ground Truth Captions	Generated Caption
1		<ol style="list-style-type: none"> 1. A man crouch on a snowy peak . 2. A man in a green jacket stand in deep snow at the base of a mountain . 3. A man kneel in the snow . 4. A man measure the depth of snow . 5. A mountain hiker be dig steak into the thick snow . 	A man with a stick in its mouth is standing on a snow covered field.
2		<ol style="list-style-type: none"> 1. A dog with a Frisbee in front of a brown dog . 2. A large black dog is catching a Frisbee while a large brown dog follows shortly after . 3. Two dark colored dogs romp in the grass with a blue Frisbee . 4. Two dogs are catching blue Frisbees in grass . 5. Two dogs are playing ; one is catching a Frisbee . 	A brown dog and a brown dog are running in a grassy field.
3		<ol style="list-style-type: none"> 1. A man is sitting on the floor outside a door and his head on his chin . 2. A man sits against a yellow wall wearing all black . 3. A man wearing a dark blue hat sits on the ground and leans against a building . 4. Man with black hat , coat , and pants sitting next to the door of a building . 5. The man in the black hat is sitting on the floor beside the green door . 	A man in a blue jacket is sitting on a city street.

Table 3. Partially successful results.





Sl. No	Image	Generated Caption
1		A group of people in a crowd.
2		A brown dog with a stick in its mouth.

Table 4. Unsuccessful results.

Sl. No	Image	Generated Caption
1		A dog with a ball in its mouth.
3		A man with a stick in its mouth with a stick in its mouth.

The classification results are shown in Table 5.

Table 5. Classification results.

Total number of images:	1000
Number of successful images:	558
Success rate:	55.8%
Number of partially successful images:	412
Partial success rate:	41.2%
Number of unsuccessful images:	30
Unsuccessful rate:	3%

The comparison of the model with various models is depicted in Table 6. The model was implemented with a beam search and game theoretic search. The proposed model with a game theoretic search achieved a BLEU score of 69.96, which was higher than all other models on the Flickr8k dataset given in Table 6. The results showed that the proposed model had a robust performance on the Flickr8k dataset.

Table 6. Comparison of the BLEU scores for different models. NIC, Neural Image Caption.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Google NIC (Vinyals et al. 2014) [3]	63.0	41.0	27.0	-
Log bilinear(kiros et al. 2014) [13]	65.6	42.4	27.7	17.7
Hard attention [43]	67.0	45.7	31.4	21.3
Soft attention [43]	67	44.8	29.9	19.5
Phi-LSTM [24]	67	44.8	29.9	19.5
Phi-LSTMv2 (w.o.r) [44]	61.5	43.1	29.6	19.7
Phi-LSTMv2 (w.r) [44]	62.7	49.4	30.7	20.8
Our Model (beam search)	67.2	55.05	44.42	40.61
Our Model (game theoretic search)	69.96	56.3	46.45	42.95

Different experiments were conducted by changing the beam size, and the BLEU scores were computed for the generated captions. The comparison of the BLEU scores for different beam sizes is given in Figure 7.

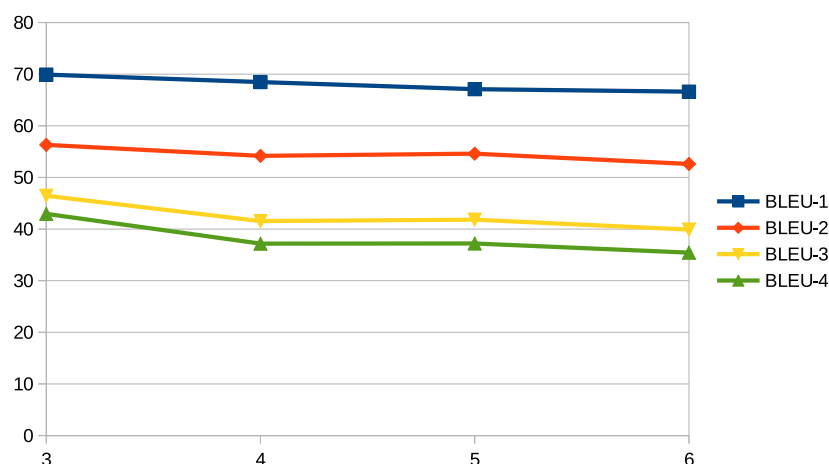


Figure 7. Comparison of the BLEU scores under different beam sizes.

The best BLEU score was obtained for a beam size of three. Therefore, the beam size was fixed to three.

6.5. Grammatical Correctness of the Generated Description

GCorrect is a new evaluation measure for monitoring the grammatical accuracy of generated descriptions. GCorrect is the mean of grammatical errors in the generated captions. It is defined by Equation (20).

$$GCorrect = \sum_{i=1}^n gerror_i / n \quad (20)$$

where *gerror* is the number of grammatical errors for each sentence and *n* is the number of sentences.

The GCorrect of this framework was **0.040625**. Grammatical errors in sentences were found using the Grammar-check package.

7. Conclusions

The proposed image captioning system had state-of-the-art performance on the Flickr8k dataset by using the BLEU score evaluation measure. In this work, a bidirectional framework for automatic image description using the densely connected convolutional neural network was developed. It considered the context of the narrative by evaluating the forward and backward analysis of trained captions. Bidirectional LSTMs gave better results for the description generation task by comparing with unidirectional LSTM. Various experiments were conducted with different deep CNNs for encoding and different RNNs for decoding. From the experimentation, Densenet was used for encoding and BLSTM for decoding. Game theoretic search and beam search were implemented for word and best caption selection. Game theoretic search was the best compared to beam search because a beam search considers the local maxima. Finally, Densenet was used for image encoding and bidirectional LSTM for caption generation in this framework. A new evaluation measure called GCorrect was proposed for measuring grammatical errors in descriptions. The system produced human readable, grammatically correct simple new sentences. The framework had better performance regarding the BLEU scores compared with state-of-the-art works. This work can be extended to generate image descriptions based on visual attention.

Author Contributions: Formal analysis, S.S.R.; Investigation, S.S.R.; Methodology, S.S.R.; Project administration, S.S.R. and S.M.I.; Software, S.S.R.; Supervision, S.M.I.; Validation, S.M.I.; Writing—original draft, S.S.R. and S.M.I.

Funding: The first author of this paper would like to thank University Grants Commission for providing fellowship through UGC NET/JRF scheme.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mikolov, T.; Karafiát, M.; Burget, L.; Černocký, J.; Khudanpur, S. Recurrent neural network based language model. In Proceedings of the Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 26–30 September 2010; DBLP, pp. 1045–1048.
2. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
3. Vinyals, O.; Toshev, A.; Bengio, S.; Erhan, D. Show and tell: A neural image caption generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
4. Karpathy, A.; Joulin, A.; Fei-Fei, L. Deep Fragment Embeddings for Bidirectional Image Sentence Mapping. *Adv. Neural Inf. Process. Syst.* **2014**, arXiv:1406.5679.
5. Bernardi, R.; Cakici, R.; Elliott, D.; Erdem, A.; Erdem, E.; Ikizler-Cinbis, N.; Keller, F.; Muscat, A.; Plank, B. Automatic Description Generation from Images: A Survey of Models, Datasets, and Evaluation Measures. *J. Artif. Intell. Res. (JAIR)* **2016**, *55*, 409–442. [[CrossRef](#)]
6. Mitchell, M.; Dodge, J.; Goyal, A.; Yamaguchi, K.; Stratos, K.; Mensch, A.; Berg, A.; Han, X.; Berg, T.; Health, O. Midge: Generating Image Descriptions From Computer Vision Detections. In Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, 23–27 April 2012; pp. 747–756.
7. Kulkarni, G.; Premraj, V.; Ordonez, V.; Dhar, S.; Li, S.; Choi, Y.; Berg, A.C.; Berg, T.L. Baby talk: Understanding and generating simple image descriptions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2891–2903. [[CrossRef](#)] [[PubMed](#)]
8. Ordonez, V.; Kulkarni, G.; Berg, T.L. Im2text: Describing images using 1 million captioned photographs. *Adv. Neural Inf.* **2011**, 1143–1151.
9. Hodosh, M.; Young, P.; Hockenmaier, J. Framing image description as a ranking task: Data, models and evaluation metrics. *J. Artif. Intell. Res.* **2013**, *47*, 853–899. [[CrossRef](#)]
10. Socher, R.; Karpathy, A.; Le, Q.V.; Manning, C.D.; Ng, A.Y. Grounded Compositional Semantics for Finding and Describing Images with Sentences. *Trans. Assoc. Comput. Linguist.* **2014**, *2*, 207–218. [[CrossRef](#)]
11. Lu, J.; Xiong, C.; Parikh, D.; Socher, R. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; Volume 6.
12. Farhadi, A.; Hejrati, M.; Sadeghi, M.A.; Young, P.; Rashtchian, C.; Hockenmaier, J.; Forsyth, D. Every picture tells a story: Generating sentences from images. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; 6314 LNCS (PART 4); Springer: Berlin/Heidelberg, Germany, 2010; pp. 15–29.
13. Kiros, R.; Salakhutdinov, R.; Zemel, R. Multimodal neural language models. In Proceedings of the 31st International Conference on Machine Learning (ICML-14), Beijing, China, 21–26 June 2014.
14. Gong, Y.; Wang, L.; Hodosh, M.; Hockenmaier, J.; Lazebnik, S. Improving image-sentence embeddings using large weakly annotated photo collections. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014.
15. Karpathy, A.; Li, F.-F. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
16. Donnelly, C. *Image Caption Generation with Recursive Neural Networks*; Department of Electrical Engineering, Stanford University: Palo Alto, CA, USA, 2016.
17. Soh, M. *Learning CNN-LSTM Architectures for Image Caption Generation*; Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, 2016.
18. Wang, C.; Yang, H.; Bartz, C.; Meinel, C. Image captioning with deep bidirectional LSTMs. In Proceedings of the 2016 ACM on Multimedia Conference, New York, NY, USA, 6–9 June 2016.
19. You, Q.; Jin, H.; Wang, Z.; Fang, C.; Luo, J. Image captioning with semantic attention. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016.

20. Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; Zhang, L. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
21. Poghosyan, A.; Sarukhanyan, H. Short-term memory with read-only unit in neural image caption generator. In Proceedings of the 2017 Computer Science and Information Technologies (CSIT), Yerevan, Armenia, 25–29 September 2017.
22. Aneja, J.; Deshpande, A.; Schwing, A.G. Convolutional image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
23. Chen, F.; Ji, R.; Sun, X.; Wu, Y.; Su, J. Groupcap: Group-based image captioning with structured relevance and diversity constraints. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
24. Tan, Y.H.; Chan, C.S. *phi-LSTM: A Phrase-Based Hierarchical LSTM Model for Image Captioning*; Springer International Publishing: Cham, Switzerland, 2017; pp. 101–117.
25. Han, M.; Chen, W.; Moges, A.D. Fast image captioning using LSTM. *Cluster Comput.* **2019**, *22*, 6143–6155. [[CrossRef](#)]
26. He, C.; Hu, H. Image captioning with text-based visual attention. *Neural Process. Lett.* **2019**, *49*, 177–185. [[CrossRef](#)]
27. Zeiler, M.D.; Rob, F. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer International Publishing: Berlin, Germany, 2014.
28. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324 [[CrossRef](#)]
29. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. In Proceedings of the NIPS, Lake Tahoe, NV, USA, 3–8 December 2012.
30. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the ICLR, San Diego, CA, USA, 7–9 May 2015.
31. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the CVPR, Las Vegas, NV, USA, 7–13 December 2015.
32. He, K.; Zhang, X.; Ren, S.; Sun, J.; Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2014.
33. Srivastava, R.K.; Greff, K.; Schmidhuber, J. Highway networks. *arXiv* **2015**, arXiv:1505.00387.
34. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. *arXiv* **2016**, arXiv:1608.06993.
35. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
36. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size. *arXiv* **2016**, arXiv:1602.07360.
37. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
38. Pennington, J.; Socher, R.; Manning, C. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Doha, Qatar, 25–29 October 2014.
39. Joulin, A.; Grave, E.; Bojanowski, P.; Douze, M.; Jégou, H.; Mikolov, T. Fasttext. zip: Compressing text classification models. *arXiv* **2016**, arXiv:1612.03651.
40. Von Neumann, O. *Morgenstern, Theory of Games and Economic Behavior*; copyright 1944; Princeton University Press: Princeton, NJ, USA, 1953.
41. Sun, X.; Liu, Y.; Li, J.; Zhu, J.; Liu, X.; Chen, H. Using cooperative game theory to optimize the feature selection problem. *Neurocomputing* **2012**, *97*, 86–93. [[CrossRef](#)]
42. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. BLEU: A method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics, Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002.

43. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. *Int. Conf. Mach. Learn.* **2015**, arXiv:1502.03044.
44. Tan, Y.H.; Chan, C.S. Phrase-based Image Captioning with Hierarchical LSTM Model. *arXiv* **2017**, arXiv:1711.05557.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).

A Comparative Study of the Performance of Different Support Vector Machine Kernels in Breast Cancer Diagnosis

Tina Elizabeth Mathew
Research Scholar
Faculty of Applied Science
University of Kerala, Thiruvananthapuram
Kerala, India
Email:tinamathew04@gmail.com

Abstract—Breast cancer is a worldwide disease affecting mostly women of all ages, race and ethnicity. Inspite of cutting edge techniques for examination and diagnosis its numbers are still on the rise. The burden of breast cancer on the healthcare system has increased steadily stressing the need for cost-effective methods for early detection, screening, and surveillance. This paper summarizes breast cancer diagnosis using support vector machines, in conjunction with other techniques, which are considered as providing the best accuracy in predicting breast cancer. The study gives an assessment on a number of papers that implement SVM to diagnose the breast cancer as well as a performance analysis of SVM models with various kernels.

Keywords—Support Vector Machine (SVM); Breast Cancer(BC), GridSearch, Data Mining, Classification

I. INTRODUCTION

Breast cancer(BC) is now the most common cancer in women especially Indian women having recently surpassed its numbers in cervical cancer. India, is witnessing more and more numbers of patients being diagnosed with breast cancer especially in the younger age group of 30-40. According to Lancet studies the projected figure of cases of Breast cancer will become almost double by 2020. Hence the need of the hour is cost effective and reliable early screening and detection techniques. Owing to its advantage of identifying decisive features from complex Breast Cancer datasets researchers have applied many machine learning techniques in improving the accuracy of breast cancer risk, detection, recurrence and survivability predictions. The selection of suitable machine learning techniques is a challenge in diagnosis of breast cancer. A review of current research reveals that almost all the ML algorithms employed in BC diagnosis are supervised. The paper reviews the various works done in diagnosing breast cancer using support vector machines and then evaluates the performance of SVM models with varying kernel functions.

The rest of the paper is organized as follows. section II summarizes the related works done with SVMs, Section III gives a description of the dataset used and Section IV compares the performance of the SVM model using different kernels Radial Basis function along with Grid search, Linear SVM with Recursive feature elimination, Polynomial Kernel with Grid Search CV, Sigmoid Kernel and Chisquare kernel. Section V gives the results obtained. The work is done using Python programming

II. RELATED WORKS IN BREAST CANCER DIAGNOSIS USING SVM

In their work Puneet[1] et al use Decision trees and Support Vector machines. The Wisconsin breast cancer dataset, with 9 attributes and 699 instances is used The training set and testing set used is of the ratio 80:20. A prediction accuracy of 90% to 94% is shown in decision trees and 94.5% to 97% accuracy in the case of SVMs. It is seen that Decision trees provided a pathway to find rules that could be evaluated for separating the input samples

into one or several groups. Huang [2] et al in their work assess the prediction performance of SVM and SVM ensembles using different kernel functions linear, polynomial, and RBF kernel functions and combination methods over small and large scale breast cancer datasets. It is seen that linear kernel based SVM ensembles based on the bagging method and RBF kernel based SVM ensembles with the boosting method are better choices for small scale datasets, where feature selection is being done in the data pre-processing stage. RBF kernel based SVM ensembles based on boosting performed better than the other classifiers for large datasets. Milon[3] et al propose a model that predicts breast cancer based on Support Vector Machine and K-Nearest Neighbors and an accuracy of 99.68% with SVM in training phase is obtained. Osman[4] in his paper proposes an automatic diagnostic method for breast tumor disease using hybrid Support Vector Machine (SVM) and the Two-Step Clustering Technique. The hybrid method enhances the accuracy by 99.1%. Bazazeh[5] compares three of the most popular ML techniques commonly used for breast cancer detection and diagnosis, namely Support Vector Machine, Random Forest and Bayesian Networks. The Wisconsin original breast cancer data set is used. The results showed that Bayesian Network has the best performance of 97.2 on an average in terms of recall and precision followed by SVM with 97% of accuracy. Random Forest technique shows the optimum ROC performance of 99.1% when compared to the two other techniques. This implies that RF has a higher chance of discriminating between malignant and benign cases. Ebrahim[6] et al in their study observed that classification implemented by Neural Network technique was more efficient compared to SVM in terms of accuracy and precision. Agarwal[7] et al in their study proposed a hybrid SVM and Logistic Regression method to improve the accuracy of prediction of survival chance of a patient post surgery. An accuracy of 85.24% was obtained, where individual methods of SVM and Logistic Regression gave an accuracy of 78.03% and 72.40% respectively. Kajitha[8] et al uses SVM and Naive Bayes technique and compares the performance of each method by measuring sensitivity, specificity, accuracy, confusion matrix and 10 fold validation. It is found that Naïve Bayes model produced a highest accuracy of 95.65% and SVM is far less accurate compared to Naïve Bayes. Hazra[9] et al devise a system to find the smallest subset of features that ensure highly accurate classification of breast cancer as either benign or malignant. A comparative study on different cancer classification approaches Naïve Bayes, Support Vector Machine and Ensemble classifiers is conducted and they measure the time complexity of each of the classifiers. Naïve Bayes classifier is considered as the best classifier with lowest time complexity when compared with the other two classifiers. Data cleaning and normalization is done and Feature selection is done by Pearson Correlation Coefficient. Sivakami[10] proposes a hybrid classification algorithm for breast cancer patients which integrated Decision Tree and SVM algorithms. Wisconsin Breast Cancer Dataset (WBCD) is used. An accuracy of 91% is obtained, k fold Crossover Validation (CV) was used to evaluate the classification accuracy. DT is implemented using Weka tool with Java platform and LIBSVM implementation for SVM. The proposed algorithm is compared with other classifiers like Instance Based Learning, Sequential Minimization Organization and Naïve Bayes. Zheng B et al.[11] develops a hybrid of K-means and support vector machine (K-SVM) algorithms. The K-means algorithm is utilized to recognize the hidden patterns of the benign and malignant tumors separately. The membership of each tumor to these patterns is calculated and treated as a new feature in the training model. It uses a support vector machine to obtain the new classifier to differentiate the incoming tumors. Based on 10-fold cross validation, the proposed methodology improves the accuracy to 97.38%, when tested on the Wisconsin Diagnostic Breast Cancer (WDBC) data set from the University of California –Irvine machine learning repository. Six abstract tumor features are extracted from the 32 original features for the training phase. Hussain[12] et al classifies the Breast cancer data set using SVM with different kernel functions- RBF, polynomial, Mahalanobis, and sigmoid and compares the results with neural networks. For feature subset selection is used employing genetic algorithms. The polynomial kernel functions of the 4th degree gives overall best performance of 92.627% and radial basis functions gives an accuracy of 92.105%. Arun[13] et al used a hybrid Decision tree SVM with reduced support vectors to speed up testing and still maintain the classification accuracy. Afay[14] in his study uses SVMs with F score and grid search to diagnose Breast Cancer. For training-testing sets of 50-50 %, 70-30% and 80-20% an accuracy of 98.53, 99.02 and 99.5% are obtained. Yue[15] et al in their paper highlighted the importance of ML techniques in healthcare applications and usage of Ensemble methods to improve the accuracy of the techniques. Bennet[16] et al in their work combined SVMs and Decision trees and found that it had better performance than individual Decision trees and SVMs. Sharma[17] et al, in their study predicts

breast cancer as benign or malignant using data sets from Wisconsin Breast Cancer Data using Logistic Regression, Nearest Neighbor and linear Support Vector Machines. Probability of recurrence in affected patients is also calculated making use of the Wisconsin Prognostic data set. It is seen that logistic regression is the best classifier model for diagnostic data set whereas support vector machine is the best classifier for prognostic data set. Wang[18] et al used a Weighted Area Under the ROC Curve Ensemble which showed better performance than single svm methods giving an accuracy of 97.10%.

III. DATASET

The Wisconsin Breast Cancer Database(WBCD) obtained from Dr. William H. Wolberg of Wisconsin University Hospitals, Madison is used. The Original data set contains 699 instances with 11 attributes each. The first attribute the ID of an instance, is discarded as it has no role in prediction, and the next 9 represent different characteristics of an instance. These are the cytological characteristics of the breast fine needle aspiration(FNA) test. The instances all have a values between 1 and 10. 1 for least problematic and 10 for the most. The diagnosis made is the last attribute. Each instance belongs to one of the 2 possible classes, benign with value 2 or malignant with value 4. The 9 attributes that are used in the prediction process are: Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, and Mitoses. The database has 699 instances with 458 benign cases - 65.5% and 241 malignant cases- 34.5%. Sixteen instances were avoided due to missing values and 683 instances(444 -benign , 239-malignant) were taken for the study.

TABLE I

Sl No	Attribute	Range of the values	Comparison of benign and malignant cells	
			Malignant	Benign
1	clump_thickness	1-10	Seen in multilayers	Seen in monolayers
2	size_uniformity	1-10	Size differs	Uniform size
3	shape_uniformity	1-10	Shape differs	Uniform shape
4	marginal_adhesion	1-10	Cells do not stick together	Cells stick together
5	epithelial_size	1-10	enlarged	small
6	bare_nucleoli	1-10	Have bare nucleoli	No bare nucleoli
7	bland_chromatin	1-10	Coarse texture	Uniform texture
8	normal_nucleoli	1-10	Nucleus is bigger	Nucleus is small
9	mitoses	1-10	Mitosis is more	Not so
	Class	2 for benign 4 for malignant		

IV. METHODOLOGY

A. Support Vector Machines

Support Vector Machine (SVM) is a supervised machine learning algorithm used for both classification and regression. It also handles multiple continuous and categorical variables. It is a binary classification method that is used with much success on large datasets and nonlinear classification problems. It is been found to be most appropriate in solving linear separated problems as well as non-linear separated problems providing very excellent accuracy rates. An SVM classifier, a concept by Vladimir Vapnik, finds the optimal separating hyperplane between positive and negative classes of data. The optimal hyperplane is the one that gives maximum margin between the training examples that lie closest to the hyperplane and the data points on the two sides belong to different classes. Support vectors are the points closest to the hyperplane. These support vectors are used to maximize the classifier margin. Different support vectors influence the position and orientation of the hyperplane. When nonlinear problems, (more than two features), are involved, SVMs map non linear inputs to a higher dimensional feature space. By defining nonlinear mappings the SVM finds an optimal hyperplane in the higher dimensional space. Symmetric functions called Kernel functions are used to define non linear mappings. Different types of kernels exist. Some commonly used kernel functions are linear, nonlinear, polynomial kernels, used in image processing, Gaussian kernel, Gaussian Radial basis function (RBF) and Laplace RBF kernels, used when there is no prior information regarding the data, Hyperbolic tangent Kernel, used in Neural Networks, Sigmoid, as proxy in Neural Networks, Linear splines kernel in one-dimension, used in large sparse data sets, Anova Radial basis Kernels, used in regression problems and Bessel function of the first kind Kernel, used to remove cross terms in functions. The most commonly used kernel for disease prediction being radial basis function. Five kernel functions are taken and studied. The work was done in python using the scikit-learn implementation of SVC.

A. SVM with Radial Basis Function kernel and Grid SearchCV

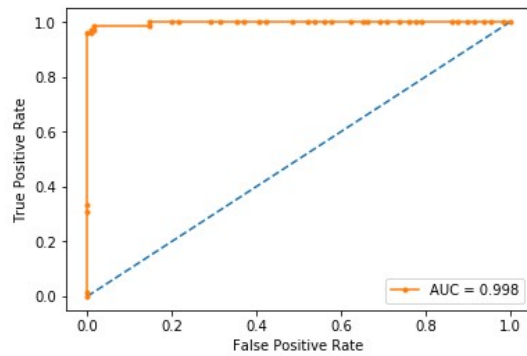
The radial basis function (RBF) kernel between two input vectors x and y is defined as $k(x,y) = \exp(-\gamma \|x-y\|^2)$ where γ , is variance. The SVM model with the Radial Basis function kernel is implemented. The dataset is normalized and partitioned into 70- 30 training -testing set. Grid search cross validation was used to find the best penalty parameter values- 'C' and 'gamma' parameter values. The model is built based on the best parameter values obtained. The data is normalized before use as normalization enhances the accuracy rates.

The performance of the classifier is shown in table I

TABLE II

Class	Precision	Recall	F1 score	Support	Accuracy	AUC	Confusion Matrix	Parameters used
2- Benign	0.98	0.99	0.98	130	0.9804	0.998	[[128 2] [3 72]]	C=1, gamma=.01
4- Malignant	0.99	0.96	0.97	75				
Weighted average scores	0.98	0.98	0.98	205				

Fig 1 ROC Curve for RBF - GSCV Kernel



B. Linear SVM with RFECV

The linear kernel, simplest of kernel functions, is represented by the equation $k(x,y)=x^T y+c_0$ for two input vectors x,y . In the experiment linear SVM was applied with recursive feature elimination with cross validation for selecting optimal number of features. RFECV used 10 fold cross validation. Feature elimination is done iteratively on the subset of remaining features excluding one attribute a time. The subset which gives the best score is taken to build the model. It was seen that the model with 6 attributes performed better and it was chosen. Features used for prediction-'clump_thickness', 'size_uniformity', 'marginal adhesion', 'bare_nucleoli', 'bland_chromatin', 'mitoses'. Features excluded-'shape_uniformity', 'epithelial_size', 'normal nucleoli'. Ranking of features- [1 4 1 2 1 3 1]

TABLE II

Class	Precision	Recall	F1 score	Support	Accuracy	AUC	Confusion matrix	Parameter used
2- Benign	0.97	0.99	0.98	130	0.9756	0.9973	[[129 1] [4 71]]	Number of Optimal Features- 6
4- Malignant	0.99	0.95	0.97	75				
Weighted average scores	0.98	0.98	0.98	205				

Fig 2 FEATURE VS ACCURACY

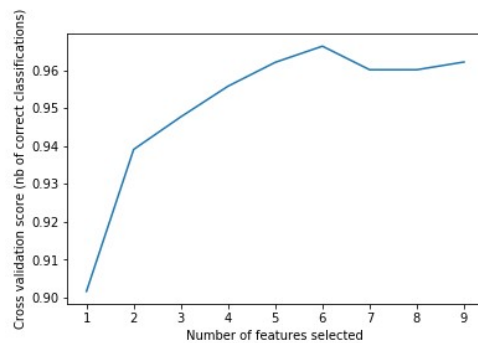
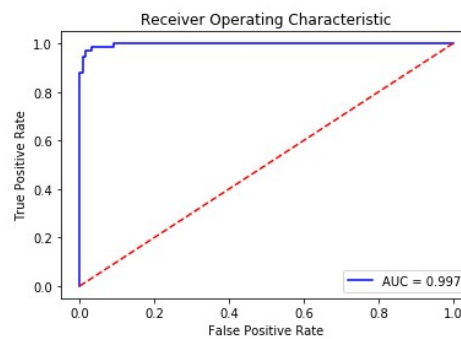


Fig 3 ROC for LINEAR SVM- RFE



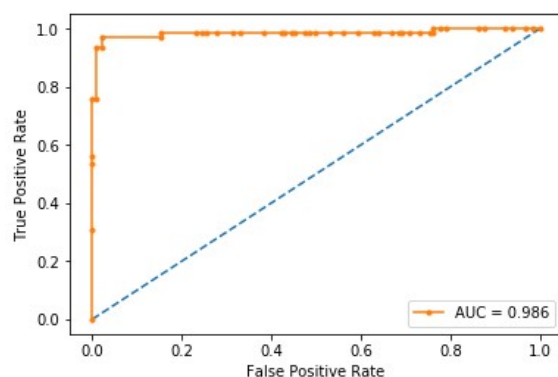
C. Polynomial Kernel with Grid SearchCV

The polynomial kernel is defined as $k(x,y)=(\gamma x^T y + c)^d$ for the input vectors x and y , d is the degree of the kernel. To find the best C and gamma values Grid Search is used. The dataset was normalized. The accuracy results and area under the curve for varying degrees of the polynomial was measured and the best accuracy and AUC was obtained at degree 5. As degree was increased above 5 the AUC illustrated a decrease in area indicating signs of over fitting.

TABLE III

Class	Precision	Recall	F1 score	Support	Accuracy	AUC	Confusion Matrix	Parameter used
2- Benign	0.86	1	0.92	130	89.27	0.986	[[130 0] [22 53]]	C=3.5, Gamma=0.11 Degree= 5 R=
4- Malignant	1	0.71	0.83	75				
Weighted average scores	0.91	0.89	0.89	205				

Fig 4 ROC of Polynomial Kernel-GSCV



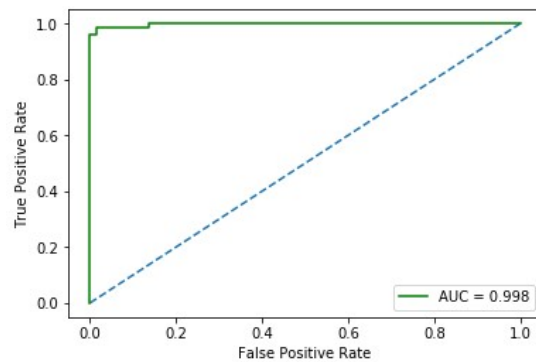
D. Sigmoid Kernel

SVM models using sigmoid kernel function is equivalent to two-layer perceptron neural networks. It is defined as $K(x,y) = \tanh(ax^T y + r)$, which takes two parameters: a , slope and r , intercept constant, for 2 input vectors x and y . Normalization of data was done. As the experiment shows it performs fairly well. Table V gives the performance value

TABLE IV

Class	Precision	Recall	F1 score	Support	Accuracy	AUC	Confusion Matrix	Parameter Values
2- Benign	0.98	0.99	0.98	130	98.05%	0.998	[[129 1] [3 72]]	C= .03 Gamma=1 r=
4- Malignant	0.99	0.96	0.97	75				
Weighted average scores	0.98	0.98	0.98	205				

Fig 5 ROC of Sigmoid Kernel

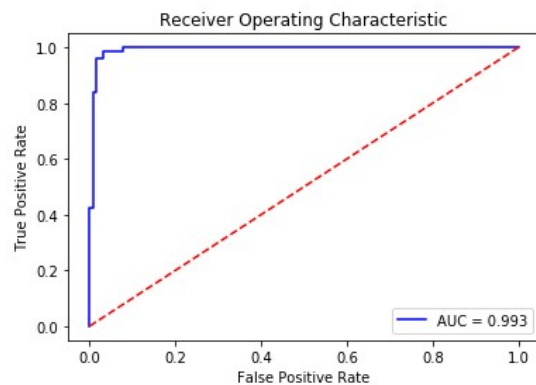


E. χ^2 kernel

The chi square (χ^2) kernel comes from the Chi-Square distribution: $K(x,y) = 1 - \sum((x_i - y_i)^2 / 0.5(x_i + y_i))$ where $i = 1$ to n . The kernel was applied in SVM to get an accuracy of 94.63. Table VI shows the performance characteristics.

TABLE V

Class	Precision	Recall	F1 score	Support	Accuracy	AUC	Confusion Matrix
2- Benign	1	0.9	0.96	130	94.63%	0.993	[[119 11] [0 75]]
4- Malignant	0.87	1	0.93	75			
Weighted average scores	0.95	0.95	0.95	205			

Fig 6 ROC of χ^2 Kernel

V. DISCUSSIONS AND RESULT

The Performance analysis of all the SVM kernels were done using the evaluation metrics Precision, Recall, Sensitivity, F1 Score and Area Under the ROC curve calculated as shown in the equations 1 to 5

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (1)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{TN}) \quad (2)$$

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{FN}) \quad (3)$$

$$\text{F1 score} = 2 * ((\text{precision} * \text{recall}) / (\text{precision} + \text{recall})) \quad (4)$$

$$\text{accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (5)$$

where TP-True Positive, FP- False Positive, TN- True Negative, FN-False Negative

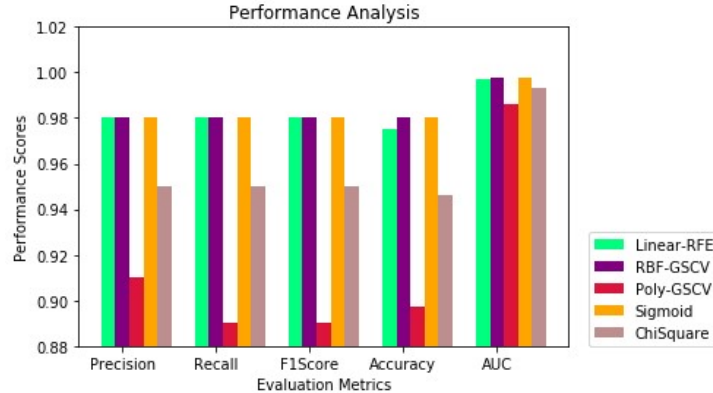
Receiver operator Characteristic(ROC) curve was drawn and the area under the curve(AUC) was calculated

The results provides understanding in the prediction performances of the various SVMs with different kernels. Linear-RFE, RBF_GSCV and Sigmoid kernels gave the best F1 score of 0.98. Recall, the ability of a model to identify all relevant instances, was lowest for polynomial kernel and highest for RBF-GSCV Sigmoid and Linear-RFE kernels. AUC is a better tool than measures such as accuracy and visualizes the performance. The higher the AUC value the better is the classifier performance. RBF-GSCV and sigmoid kernels showed the largest AUC value. It can be concluded from the results that the RBF, Sigmoid and Linear-RFE kernels are better for breast cancer diagnosis prediction.

TABLE VI COMPARISON OF KERNELS

Sl. No	SVM Kernel	Precision	Recall	F1 Score	Accuracy%	AUC
1	Linear-RFE	0.98	0.98	0.98	97.5	0.998
2	RBF-GSCV	0.98	0.98	0.98	98.0	0.998
3	Polynomial-GSCV	0.91	0.89	0.89	89.2	0.986
4	Sigmoid	0.98	0.98	0.98	98.0	0.998
5	Chisquare	0.95	0.95	0.95	94.6	0.993

Fig 7 PERFORMANCE ANALYSIS



VI. CONCLUSION

Different ML approaches employed in BC diagnosis and prognosis is analyzed using the WBCD database as a benchmark dataset. ML techniques have shown their remarkable ability to improve classification and prediction accuracy. However, in biomedical sciences where accuracy of predicted outcomes are of at most importance, and since the methods vary for different datasets, it is vital that these techniques which are difficult to comprehend, are examined and assessed rigorously before using them commercially. The performance of SVM with different kernel functions were assessed. The performance of SVM in combination with other methods or as an ensemble is to be analyzed in future.

ACKNOWLEDGMENT

The author is thankful to all those who gave valuable inputs and support for the work. The breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

REFERENCES

- [1] Puneet Yadav, Rajat Varshney, Vishan Kumar Gupta, Diagnosis of Breast Cancer using Decision Tree Models and SVM, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 05 Issue: 03 | Mar-2018.
- [2] Min-Wei Huang, Chih-WenChen, Wei-ChaoLin, Shih-WenKe, Chih-FongTsai, SVM and SVM Ensembles in Breast Cancer Prediction. PLOSONE|DOI:10.1371/journal.pone.0161501 January 6,2017
- [3] Md. Milon Islam, Hasib Iqbal, Md. Rezwanul Haque, and Md. Kamrul Hasan, Prediction of Breast Cancer Using Support Vector Machine and K-Nearest Neighbors, 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC) 21 - 23 Dec 2017, Dhaka, Bangladesh
- [4] Ahmed Hamza Osman, An Enhanced Breast Cancer Diagnosis Scheme based on Two-Step-SVM Technique, International Journal of Advanced Computer Science and Applications, Vol. 8, No. 4, 2017, page 158
- [5] Dana Bazazeh and Raed Shubair, Comparative Study of Machine Learning Algorithms for Breast Cancer Detection and Diagnosis, International Journal of Trend in Research and Development, Volume 4(3), ISSN: 2394-9333, May-June 2017
- [6] Ebrahim Edriss Ebrahim Ali, Wu Zhi Feng, Breast Cancer Classification using Support Vector Machine and Neural Network, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064, Volume 5 Issue 3, March 2016
- [7] Rakshanda Agarwal, Rajeshkannan Regunathan, A hybrid SVM method for survival of patient post breast cancer operation prediction by using SVM and logistic regression, International Journal of Engineering & Technology.
- [8] Kathija1, Shajun Nisha2, Breast Cancer Data Classification Using SVM and Naïve Bayes Techniques, International Journal of Innovative Research in Computer and Communication Engineering. Vol. 4, Issue 12, December 2016
- [9] Animesh Hazra, Subrata Kumar Mandal, Amit Gupta, Study and Analysis of Breast Cancer Cell Detection using Naïve Bayes, SVM and Ensemble Algorithms, International Journal of Computer Applications (0975 – 8887) Volume 145 – No.2, July 2016.
- [10] Sivakami, "Mining Big Data: Breast Cancer Prediction using DT-SVM Hybrid Model." International Journal of Scientific Engineering and Applied Science (IJEAS) -Volume-1, Issue-5, ISSN: 2395-3470, August 2015
- [11] Bichen Zheng, Sang Won Yoon, Sarah S. Lam, Breast cancer diagnosis based on feature extraction using a hybrid 4 of K-means and support vector machine algorithms, Expert Systems with Applications (2013)
- [12] Muhammad Hussain, Summrina Kanwal Wajid, Ali Elzaart, Mohammed Berbar, A Comparison of SVM Kernel Functions for Breast Cancer Detection, 2011 Eighth International Conference Computer Graphics, Imaging and Visualization
- [13] M. Arun Kumar, M. Gopal, A hybrid SVM based decision tree, Elsevier Pattern Recognition 43 (2010) 3977–3987
- [14] M.F. Akay, Support Vector Machines combined with feature selection for breast cancer diagnosis/Expert Systems with Applications 36 (2009) 3240–3247.
- [15] Wenbin Yue 1 ID, Zidong Wang 1,*, Hongwei Chen 2 ID and Annette Payne 1 and Xiaohui Liu, Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis, Designs 2018, 2, 13
- [16] A Support Vector Machine Approach to Decision Trees, K.P. Bennett, J.A. Blue
- [17] Machine Learning Approaches for Breast Cancer Diagnosis and Prognosis Ayush Sharma, Sudhanshu Kulshrestha, Sibi Daniel, In Proceedings of the International Conference on Soft Computing and Its Engineering Applications, Changa, India, 1–2 December 2017.

[18]Haifeng Wang, Bichen Zhenga, Sang Won Yoona, Hoo Sang Kob, A Support Vector Machine-Based Ensemble Algorithm for Breast Cancer Diagnosis, European Journal of Operational Research December 6, 2017

[19]. O. L. Mangasarian and W. H. Wolberg: "Cancer diagnosis via linear programming", SIAM News, Volume 23, Number 5, September 1990, pp 1 & 18.

[20] William H. Wolberg and O.L. Mangasarian: "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp 9193-9196.

[21] O. L. Mangasarian, R. Setiono, and W.H. Wolberg: "Pattern recognition via linear programming: Theory and application to medical diagnosis", in: "Large-scale numerical optimization", Thomas F. Coleman and Yuying Li, editors, SIAM Publications, Philadelphia 1990, pp 22-30.

[22] K. P. Bennett & O. L. Mangasarian: "Robust linear programming discrimination of two linearly inseparable sets", Optimization Methods and Software 1, 1992, 23-34 (Gordon & Breach Science Publishers).

Tina Elizabeth Mathew is a PhD student in the Faculty of Applied Sciences at the University of Kerala, Thiruvananthapuram, Kerala, India. She is an Assistant Professor in the Department of Computer Science, Government College Kariavattom at Thiruvananthapuram, Kerala, India with a teaching experience of 15 years. She has completed her Masters in Computer Science from School of Computer Sciences at Mahatma Gandhi University, Kottayam, Kerala, India in 1995 and her Bachelor of Science in Mathematics in 1993 from Mar Thoma College, Tiruvalla affiliated to Mahatma Gandhi University, Kottayam. Her current research interest are in Data Mining and Machine Learning.



Breast Cancer Diagnosis using Stacking and Voting Ensemble models with Bayesian Methods as Base Classifiers

Tina Elizabeth Mathew¹, K S Anil Kumar², K Satheesh Kumar³

¹Research Scholar, Faculty of Applied Science and Technology,

²Research Guide, Technology Management, Faculty of Applied Science and Technology,

³Research Guide and Head, Future Studies, Faculty of Applied Science and Technology

¹University of Kerala, Thiruvananthapuram, Kerala, India

²University of Kerala, Thiruvananthapuram, Kerala, India

³University of Kerala, Thiruvananthapuram, Kerala, India

¹Email:tinamathew04@gmail.com

²Email:ksanilksitm@gmail.com

³Email: ksktvm@gmail.com

Abstract- Breast Cancer is a deadly disease affecting mostly women. Despite of many modern techniques Breast Cancer is still alarmingly on the rise and the diagnosis still needs to be improved for timely identification of the disease. A plethora of machine learning techniques have been used in disease diagnosis, risk, recurrence and survivability predictions in the yesteryears, Machine learning models have been used individually and as ensembles for classification and prediction in the medical field. Bayesian Methods provide better classification performance, interpretability and assist in constructing inferences in uncertain conditions. In this study the classification accuracy of single Bayesian Methods- Naive Bayes, Hidden Naive Bayes, Bayesian Belief Networks- in breast cancer diagnosis is analyzed. Bagging, Dagging and Boosting ensemble techniques are applied in conjunction with the Bayesian methods and analyzed. Two Models are proposed . A Stacking model and Stacking-Voting model. In the Stacking model the 3 Bayesian approaches in conjunction with Logistic Regression and Sequential Minimal Optimization as base classifiers and REPTree as meta classifier is used and, in Stacking and Voting - the 3 Bayesian Methods with 2- meta classifiers REPTree and Random Forest , Bayesian Network with Stochastic Gradient Descent and a 2 -meta classifier of REPTree and Decision Stump is proposed and the model performances are evaluated. It was seen that the new proposed ensembles had a better performance than the other models in most cases.

Keywords: Naive Bayes(NB), Hidden Naive Bayes(HNB), Bayesian Networks(Bayes Net), Breast Cancer, Ensembles, Logistic Regression(LR), Stochastic Gradient Descent(SGD), Reduced Error Pruning Tree(REPTree), Decision Stump

I. INTRODUCTION

Breast Cancer etiology is very complex. A variety of factors can be associated with its incidence which influences prediction accuracy. There are various techniques to diagnose breast cancer such as auto exploration, mammography, FNA, ultrasound, MRI and thermography. Despite of progress in diagnostic methods Breast cancer is still one of the leading cancers among women worldwide. Statistical and machine learning techniques can be used in combination as complementary methods in aiding early and better diagnosis. Bayesian methods have found to be effective in dealing with health care evaluation, identifying therapeutic targets[1], risk prediction of breast cancer recurrence [2] and many more.

Classification techniques like decision tree, regression, SVM, ANN have been widely used in the medical field for disease susceptibility prediction, survivability prediction and recurrence prediction [16]. Bayesian approaches are simple robust methods, with wide applicability, using conditional probabilities that help to combine new information with existing information. It helps in better modeling data with uncertainties and missing values.

Bayesian theory is named after the 18th century British mathematician Thomas Bayes. Although he introduced it in 1763 it gained importance in the 1950s and 1960s. Langarizadeh et al [11]in their systemic review of PubMed articles between 2005 and 2016 found in, 23 studies and 53,725 patients, that for predicting disease naive Bayesian networks were used and were effective than other methods such as SVM, Logistic regression, decision trees, ANN, TANs etc. . It can be concluded from various studies that that predictive models are significant approaches in clinical practice and diagnosis. Bayesian methods create a model that links data with parameters and are well suited for decision making. Uncertainties occur due to lack of knowledge of relevant facts. Bayesian methods can quantify these uncertainties using its probability, which is very crucial for decision making. Bayesian models can be combined with more complex models to identify all facts that affects the final results. Studies show that kernel-based classifiers are optimal, and hence Bayes methods, are the most prudent to be chosen as ensemble classifiers[23].

Despite the of success of traditional machine learning methods, many a times they may fail due to noisy data, imbalanced data, complex data and high dimensional data. Hence it is necessary to construct highly efficient models. Ensemble models are one of such methods that improve over traditional methods. Ensemble methods also known as committee based learners or learning multiple classifier systems, are machine learning techniques which combine many base or weak models and produce an optimal predictive model. Most of the ensembles usually use single base learners to give homogenous ensembles. Multiple base learners can be used to form heterogeneous ensembles[3,32].

The remainder of this paper is organized as follows. The next section describes the related works. Section II presents the data used. Section III illustrates the Methodology used. Section IV gives the results and discussion and finally the conclusion is drawn in section V.

A. Related Works

Tribhuvan et al[4] in their paper compared ID3 Decision trees and Naive Bayes using student performance data from an educational institution in Weka. They concluded that ID3 provided better accuracy performance and Naive Bayes had a higher error rate with an accuracy of 92%. Aidaroos et al[5] in their study made an overview on medical data mining. They compared Naive Bayes against 5 other classifiers -Logistic Regression, kstar, Multilayer perceptron, J48, ZeroR using Weka and the 15 UCI repository medical datasets. They found that naive bayes was better in terms of accuracy in most datasets. Nugroho et al[6] proposed a cascade generalization of four Bayesian Network based methods, SVM, and C4.5 to predict breast cancer. It was compared with bagging and individual classifiers and was seen to perform better. Mamographic mass data of 961 instances and 6 attributes with 516 benign instances and 445 malignant instances was used in this study. The highest ROC area under curve of 0.903 was shown by Naive Bayes with SMO cascade. and the highest accuracy of 83.689% by Bayesian Network using Tabu search with SMO cascade.

Karabatak[7] proposed a weighted naive bayes classifier it produced an accuracy of 98.54%. A grid search mechanism was used to find optimal weight for the attributes. Grid search has a high computational cost and the initialization of weights is crucial and dependent on the application. Lopez[8] used Naive Bayes, Tree Augmented Naive Bayes, K-dependence Bayesian classifier and Forest Augmented Naive Bayes for breast mass classification task. They experimented on two datasets of BCDR-F01 database, having 112 benign and 119 malignant masses using the full feature sets and with a subset of 8 features from the feature sets. Fischer score was used to select best features. They used Matlab with help of the Bayes Net toolbox and BNT Structure Learning Package. Maysanhaya[9] proposed a hybrid machine learning method to classify the types of breast cancer by using a combination of Wrapper and Naive Bayes algorithms. Banu et al[10] in their work reported the performance of Bayes classifiers like Tree Augmented Naive Bayes (TAN), Boosted Augmented Naive Bayes (BAN) and Bayes Belief Network (BBN). They concluded that TANs provided the best accuracy. the tool used was SAS -EM (Statistical Analytical Software Enterprise Miner). Guzel et al[12] in their approach used k Nearest Neighbor algorithm (kNN) and Naive Bayes to impute missing values. Then, the performance of the system is evaluated by kNN and Naive Bayes classifiers to detect breast cancer. Rathi et al[13] in their study compared Naive Bayes with SMO, J48, Bayesian Networks and found that Naive Bayes was much better in performance, 97.5%, using Java Net Beans Interface. the 699 instances of the WBC dataset from UCI repository was used. Data cleaning was applied and records with missing values were substituted with values nearest to them. Udayakumae et al[14] used Bayes algorithm and neural networks (NNs), to identify the type of the mammogram and stages. A good accuracy for classification was obtained. Sesen[15] used Bayesian Networks in survivability prediction and treatment selection recommendation of Lung Cancer The English Lung Cancer database (LUCADA), having 126,000 patients who were diagnosed between 2006 and 2010 was used. To improve the model structure learning was applied using the CAMML hybrid causal discovery algorithm.

Yeulkar et al[16] in their work compared three Naive Bayes algorithms and C4.5 algorithm to classify malignant and benign tumors using the SEER dataset. The Naive Bayes algorithm with laplace and metric computation was used. C4.5 was seen to have better performance of 98.09 % accuracy. Nahar[17] in their work proposed a kernel based naive bayes classifier to classify breast tumour in mammography data. They compared their work with C4.5 and Kernel based c4.5. Kernel based naive bayes produced better results. Umamaheswari[18] compared 5 classification algorithms Bayesian Networks, Naive bayes, Decision Trees, J48, ADTree, on wisconsin breast cancer dataset and found that Bayesian Networks were the best. T test was used to find the better algorithm. Demigha et al[19] proposed a methodology for mining medical knowledge based on the Bayesian Classification to predict and detect anomalies in breast cancer. They demonstrated the suitability of Naive Bayesian methods and weighted naive bayesian classifier in classification of breast cancer. Chang et al[21] used Bayesian Logistic Regression for breast cancer prediction and analyzed its performance. The Wisconsin diagnostic data of 569 instances and 32 variable was used.

Soria et al [22] used three classifiers - C4.5, Naive Bayes, multilayer Perceptron on a dataset of 25 tumour markers with 663 instances for breast cancer. When 10 tumour markers were used the performance increased in all classifiers. MLP was the best classifier but Naive Bayes showed better performance with a reduced feature set. Joshi et al [24] in used web usage mining for finding hidden patterns in the breast cancer dataset. They compared 37 classifiers and It was seen that alongwith the various methods Naive Bayes showed a good classification accuracy. Kharya et al [25] in their proposed system used Bayesian Networks in an automated breast cancer detection support tool and conclude that it was suitable for mammographic decision support. They implemented two Bayesian Networks on the mammographic mass dataset; tree augmented Naive Bayes and Markov blanket estimation learning algorithms. Bayesian Networks were compared with Multilayer Perceptron and were found much better. A Bayesian Network was created to detect cancer malignancy using Bayesian Net Generator. Abdara [26] used a stacking Voting Nested ensemble to create an automated diagnosis system for breast cancer. They used Naive Bayes and Bayesian Networks as single base classifiers. Two layered nested ensembles with 2 and 3 meta classifiers were used. Wisconsin Diagnostic Breast Cancer (WDBC) dataset and K-fold Cross Validation technique was used for evaluating the model. Comparison of the proposed model was done with individual Naive Bayes and bayesian Network models and the proposed model achieved an accuracy of 98.07%.

Ibeni et al [27] in their paper used naïve bayes (NB), bayesian networks (BN), and tree augmented naïve bayes (TAN) with three datasets; Breast cancer, breast cancer wisconsin, and breast tissue dataset. The classifiers were compared with K-nearest neighbor (K-NN), support vector machine (SVM), and decision tree (DT). They obtained an accuracy % of 97.281 using Bayesian Networks. Ratnawali et al [29] in their work used a Modified K-Means Naïve Bayes (KMNB) method on Breast Cancer data. An accuracy of 95% was received. The modification was a method change to configure the initial centroid used in the original k means method. They used k means to classify data into 3 clusters- benign, malignant and maybe. The naive bayes method was then applied on the maybe class. The hybrid method was considered better in prediction accuracy. Prediction error was reduced by 50% but had longer computation time. Kotsianti et al [31] in their work built an ensemble using a voting methodology of bagging, boosting and dagging ensembles using 8 sub-classifiers in each sub ensemble. They used sum rule voting for combining bagging, dagging and boosting. 36 datasets were used for this study. Decision Stump was the base classifier. It was seen that this ensemble was better almost all cases and was better than individual ensembles. Ji et al [34] in their work proposed a combination model called packaged hidden naive bayes which combined naive bayes and Hidden Naive Bayes classifiers. The new model showed better performance than the individual models. Ganpati et al [35] in their study analyzed three meta classification algorithms END- Ensemble of nested dichotomies, bagging and dagging on a data set from UCI repository with 20000 instances and 17 attributes. They found the END algorithm to be more efficient in terms of accuracy, 94.86% and error rate. Weka toolset was used. Kathija et al [36] in their paper compared Naive Bayes and SVM using the WBCD Dataset. They found Naive Bayes, 95.65% to provide better accuracy performance. The work aims to find the smallest subset of features from Wisconsin Diagnosis Breast Cancer (WDBC) dataset by applying confusion matrix accuracy and 10-fold cross validation method that will ensure accurate ensemble classification of breast cancer into either benign or malignant.

II DATA

The University of Wisconsin Breast Cancer Database available from University of California, Irvine (UCI) machine learning repository is being used in this study. The dataset was collected from Madison Hospital and created by Dr William H Wolberg between 1989 and 1991. The Wisconsin Breast Cancer dataset consists of 699 instances with missing values for 16 instances. It has two classes benign and malignant denoted by values 2 and 4 respectively. The data set consists of 10 attributes; which are sample code number, clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. The 683 instances, out of which 444 are benign cancers and 239 are malignant cancers, are taken for study. The 16 instances with missing values are avoided.

A. Preprocessing

The dataset is converted to the arff format which is used in WEKA. The numeric data values are converted to nominal values using the weka filter numericToNominal. The breast cancer dataset is then divided into training and testing sets based on 10-fold validation method where the nine sets are used for training the algorithm and the last one set for testing and assessing the algorithm.

III. METHODOLOGY USED

Various Ensembling techniques exist. In this study, Bagging, Dagging Boosting, Stacking and Voting ensembles are used. The choice of base and meta classifiers is a decisive factor for the success of the prediction model. The classifiers used in the ensembles are explained in the ensuing subsections.

A. Bayesian Belief Networks

Bayesian Networks also known as belief networks, consists of a directed acyclic graph (DAG), and a joint probability distribution. . It is a category of Probabilistic Graphic modeling and models uncertainties using Probabilities. A DAG consists of nodes and links. Its node represents attributes and the arcs correspond to attribute dependencies or relations between nodes[26]. The links have directions. The conditional probability of each node or attribute is computed by the classifier. For two nodes X and Y and if they are connected and the Probability of X is given by P(X) and conditional probability by P(X/Y) then the joint probability of all X's is given in equation 1.

$$P(X_1, X_2, \dots, X_n) = \pi P(X/P(X_i)). \quad \text{Equation (1)}$$

Bayesian network classifiers are advantageous in that they have the ability to deal with missing values, explicitly provide the conditional probability distributions of the values of the class attributes given the values of the other input attributes and they are easy to comprehend[25].

B. Naive Bayes

Naive Bayes is a Classification algorithm working on the basis of Bayes Theorem. It is a probabilistic classifier which takes the simplest form of Bayesian Networks. For a sample X, and a hypothesis H, P(H|X) which is the probability that given X, the hypothesis H is true, then Bayes theorem is as shown in equation 2.

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \quad \text{Equation (2)}$$

It can be considered as an independent feature model as it assumes that the lack/existence of a feature is independent of the lack / existence of another feature. This can be a disadvantage too. Naive bayes needs only a small amount of training data to estimate the parameter necessary for classification[14]

C. Hidden Naive Bayes

In an HNB, a hidden parent is created for each attribute, which combines the influence from all other attributes. It is a model which avoids the complexities of Bayesian network structures[28].

D. Logistic Regression(LR)

Logistic regression is a statistical and predictive analysis technique invented by Dr Cox in 1958 that relates the dependent variables with independent variables. The dependent variables are the output variable or class of the data eg. benign or malignant class in case of disease prediction. The independent variables are the attributes available in the dataset. Logistic Regression makes use of the logistic or sigmoid function. and is also known as log odds ratio or logit function. The coefficients of the logistic regression algorithm is estimated from the provided training data using the maximum-likelihood estimation method, which finds the best value for the coefficients of the data. The logistic function is defined in equation 3. The logistic model is appropriate when the output is binary and gives probabilities but it suffers from complete separation and is vulnerable to overfitting.

$$\sigma(t) = \frac{1}{1+e^{-t}} \quad \text{or} \quad \ln\left(\frac{P}{1-P}\right) = t \quad \text{Equation (3)}$$

E. Sequential Minimal Optimization SVM

The SMO algorithm is used for training support vector machines . It was developed by John C Platt in 1998 and is the fastest quadratic programming optimization algorithm. The SMO algorithm is easy to implement when compared to traditional SVM algorithms. It minimizes memory storage, has high accuracy and faster training time. It can handle very large training datasets. SMO subdivides the problem into small sub problems and solves them. The Sequential Minimal Optimization (SMO) algorithm uses the decomposition method and optimizes a minimal subset of just two points at each iteration by enforcing the

condition $\sum y_i \alpha_i = 0$. At each step SMO chooses two elements α_i and α_j , lagrange's multipliers, to be jointly optimized. All the others are fixed, and the α vector is updated accordingly. The lagrange's multipliers help in finding the support vectors for the svm. A heuristic like hill climbing is used to choose the optimal α values.

F. Stochastic Gradient Descent(SGD)

Stochastic Gradient Descent is an optimization algorithm that finds parameter values of a function such that the cost function is minimized. It is a variation of the Gradient Descent method. The main aim is to find the best estimate for a target function (f) that maps the input data (X) onto the output variables (Y). It calculates the derivative from each training data instance and calculates the update immediately. It is a suitable method for large datasets as it needs only small number of iterations for finding solutions besides having lower computational cost.

G. Reduced Error Pruning Tree (REPTree)

REPTree is a fast decision tree classifier that creates multiple trees in different iterations and then selects the best one from the generated trees. This will be taken as the model for classification. Information gain and entropy which minimized the variance is calculated for selecting attributes. In pruning the tree the mean square error measure is used on the predictions made by the tree. If H represents entropy, using entropy of H and conditional entropy of Y, where $Y = [y_1, y_2, \dots, y_n]$ and $X = [x_1, x_2, \dots, x_n]$ are discrete variables, Information Gain(IG) is given as in equation 4.

$$IG(Y;X) = H(Y) - H(Y|X)$$

Equation (4)

H. Decision Stump

Decision Stump is a one level decision tree that uses one attribute for splitting. It is seen to produce efficient results when used in boosting algorithms. Each node in a decision stump represents a feature in an instance to be classified, and each branch represents a value that the node can take. Instances are classified starting at the root node and sorting them based on their feature values[31].

I. Ensemble methods

Ensemble methods are combination of machine learning techniques. They are seen to provide better and stable predictions when pitted against individual methods, The following are a few ensemble methods used in this study.

1. Bagging

Bootstrap Aggregating or Bagging, proposed by Leo Breiman in 1994, is an ensemble method that improves the performance of classifiers. It reduces variance and prevents overfitting. For a dataset D of size n, bagging produces m subsets of D, (by sampling D with replacement), of size n which are used as training sets. While choosing samples some instances may not be included while some may be used repeatedly. When replacement is not used it is called pasting. The classification done by these randomly generated training sets on a base classifier are combined, by voting for the majority value, and it is taken as the classification of the model. Bagging is seen effective for unstable methods. Computations to fit the models are done in parallel.

2. Dagging

Dagging takes a number of disjoint subsets from the dataset and each of these subsets are given to the base learner. Final predictions are made with a majority vote. Bagging and Dagging are seen better for noisy data[31].

3. Boosting

Boosting algorithms introduced by Schapire in 1990 are considered stronger than Bagging and Dagging in the case of noise free data[31]. Computations are similar to that of bagging except that learning is done sequentially correcting its predecessor. Boosting models train the classifiers sequentially and learn from previous mistakes. The weights of misclassified data are increased and that of correct instances are reduced and fed to the next classifier. Boosting is considered to give better performance than bagging but it tends to over fit. To avoid overfitting 10 fold cross validation can be used. Weka has a few boosting methods like Adaboost, Multiboost and Realboost with Adaboost being the most popular one.

4 . Stacking

Stacking introduced by Wolpert in 1992 works in two steps. It uses heterogeneous weak learners to predict a class and then these predictions are combined using a meta classifier. In stacking the training data is split into k folds. The weak learners are trained on $(k-1)$ folds and tested on one fold. This is done iteratively for all the folds so that predictions and testing are obtained for all the folds. The meta model results are computed based on the outputs of the weak learner models. It combines the predictions by reducing the generalization errors.

5.Voting

Voting ensemble is a meta classifier that combines similar or dissimilar classifiers for classification. Voting are of two types hard voting and soft voting. In hard voting the final prediction is done based on the majority output of the base classifiers whereas soft voting takes the average of the probability predictions of the base classifiers.

J. Random Forest

Random Forest introduced by Breiman in 2001, is a supervised learning technique that consists of many individual decision trees that work as an ensemble. It uses a divide and conquer approach and splits the data set into random sets. A tree each is generated for each randomly split data subset. Each individual tree provides a class prediction and the class with most votes are used as the model's prediction. The more the decision trees the better is the prediction accuracy. The advantage of this model is that it does not overfit and can be used for categorical values as well. It handles missing values efficiently. Even though Random Forests are robust and low cost models they can be time consuming. Random Forests are similar to bagging except that they use a random selection of features for finding the best node for splitting while bagging uses the whole feature set. Each tree in random forest is split based on different features while the split in bagging is on the same features. The pruning done by Random Forest reduces the chance of overfitting. Random forest is an extended version of bagging and has many properties of bagging. They improve the variance reduction by reducing the correlation between the trees[23].

K. Proposed Method

Two methods are proposed one using stacking{S} alone and another using stacking and voting (S-V) ensembles. A few classifiers are used as base and some as meta classifiers. In the first proposed model stacking alone is used with the three Bayesian methods Naive Bayes, Hidden Naive Bayes and Bayesian Network as base classifiers. Each of the methods are combined with two other classifiers- Logistic Regression and Sequential Minimal Optimization- creating separate 6 models- Naive Bayes+ Logistic Regression(NB+LR), Naive bayes+SMO-(NB+SMO), Hidden Naive bayes+ Logistic regression(HNB+LR), Hidden Naive Bayes+ SMO -(HNB+SMO) and Bayesian Networks+ Logistic Regression(BayesNet+LR), Bayesian Networks+SMO- (BayesNet+SMO). The Meta classifier used in all cases is REPTree and the performance of each of the six models are compared.

The working of the proposed stacking model is explained using two levels in this section. Initially the data set is partitioned into 10 folds, 9 folds for training and 1 fold for testing. In the first level the two base classifiers are trained and tested on the 10 folds. The 10 models on the used bayesian approach and 10 models of the used base predictor(LR/SMO) are produced. These predictions are averaged and the new predictions of each model known as meta features is stored and is later used by the meta classifier REPTree as an additional feature. In the next level the meta classifier, REPTree, trains and tests on the dataset along with the predicted outcomes of the base models to produce the final predictions.

In the second proposed model, stacking and voting(S-V) is applied. A general approach of the proposed model is that in stacking it uses 2 base classifiers and as meta classifier voting technique is used. Voting combines two classifiers. A few classifier combinations are used in this study to examine performance of the Stacking-Voting model. The 3 Bayesian methods - NB, HNB, BayesNet are used individually with SMO as base classifiers and a voting ensemble of REPTree and Random Forest as meta classifiers.

In another combination, Bayesian Networks are used with Stochastic Gradient Descent as weak learners(Bayesian Networks + SGD) and a 2- meta classifier using voting with REPTree and Decision Stump. The StackingC and voting function in weka is used. Average of the probabilities of the individual classifiers is used for combining the probabilities of the classifiers and obtaining the probability of the new voting model(Soft Voting is used). The generalized approach of both the proposed models are shown in figures 1 and 2 respectively.

In the S-V model, initially stacking is applied to the two base classifiers. It is done in 2 levels. The dataset which is partitioned to 10 folds of which 9 is used for training and 1 for testing. The two base classifiers are trained on these folds repeatedly. Ten models each are created against both of the classifiers. The output of each classifier is averaged and taken. The dataset is updated with the predictions made by each of the classifiers. The meta features are included in the original dataset.

In the next level Voting ensemble is used as a meta classifier. Two classifiers used here are in one case 1) Random Forest and REPTree and in the other case 2) Random Forest and Decision Stump. In voting each model is trained and tested separately. The predictions are then combined using average of probabilities for obtaining the final prediction. The block diagrams of the Stacking and Stacking-Voting models are shown in figures 1 and 2 respectively.

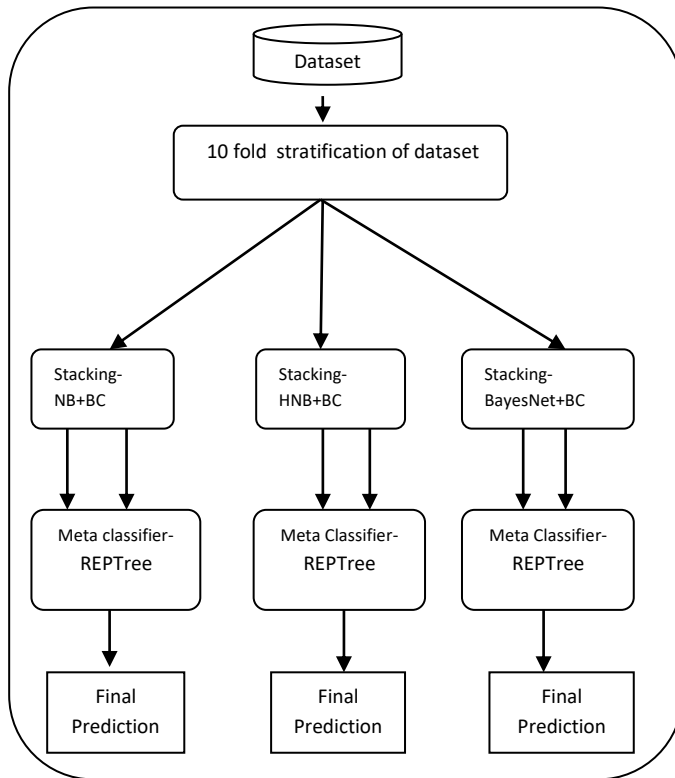


Fig.1 Stacking Model

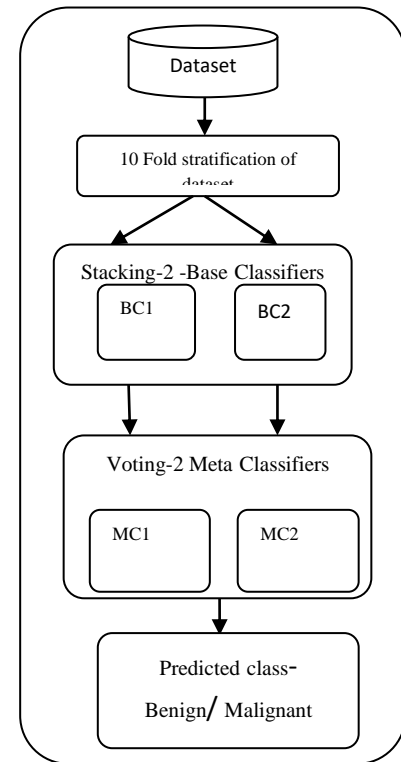


Fig. 2 Generalized Stacking - Voting Model

IV. RESULTS AND DISCUSSION

A. Accuracy Measures Used

The accuracy of individual methods of Naive Bayes, Hidden Naive Bayes and Bayesian Networks are 97.36, 96.04, and 97.65 respectively as shown in table 1. When ensemble techniques of bagging, dagging and boosting were done the results obtained are shown in table 2. Naive Bayes with bagging produced a better result of 97.5 while Naive Bayes with dagging produced 96.77 and with boosting accuracy was 97.36%, same as individual Naive Bayes. Bagging was seen effective with Naive Bayes. In the case of Bayesian Networks, Bagging and Boosting produced an accuracy of 97.5% each, while dagging produced slightly lower accuracy of 96.92%. Ensembling Hidden Naive Bayes seemed to reduce the accuracy. On the large Bagging, Dagging and Boosting with HNB lowered the accuracy and among the three methods bagging was comparatively better followed by boosting. The accuracy performance, with bagging was 95.6%. Dagging produced a comparatively low score of 92.67 and boosting produced 94.72% accuracy. Dagging on all three bayesian methods was seen to lower the accuracy%. Accuracy measures of individual and bagging-dagging-boosting ensembles are in Table I and Table II respectively.

In the proposed method using Stacking alone it was seen that Naive Bayes combinations with Logistic Regression and SMO gave the best prediction accuracy of 97.8% each, followed by Bayesian Networks, 97.2% each, and Hidden Naive Bayes

with 96.4% and 96.9% . Time required for building stacking models with Logistic regression was more when compared with stacking models using SMO.

In the Stacking and Voting model of NB+ SMO and Bayes Net+ SMO with voting meta classifiers REPTree and Random Forest, performance accuracy was 97.8 and 97.2%, similar to the Stacking only models but the time taken for building models was reduced considerably when compared with that of the stacking model using LR. The stacking model using SMO and the stacking-voting models were almost similar in time usage. The accuracy measures of stacking ensembles are shown in Table III. Bayesian Networks with SGD used voting meta classifiers with REPTree and Decision Stump and an accuracy of 97.6 was obtained. It was better than the S-V ensemble using Bayes Network with SMO.

TABLE I ACCURACY OF INDIVIDUAL MODELS

Classifier	Correctly Classified Instances	Accuracy	Inaccurately Classified Instances	Inaccuracy	Confusion Matrix
Naive Bayes	665	97.3646	18	2.6354	431 13 5 234
Hidden Naive Bayes	656	96.0469	27	3.9531	436 8 19 220
BayesNet	667	97.6574	16	2.3426	431 13 3 236

TABLE II ACCURACY OF BAGGING - DAGGING- BOOSTING MODELS

Classifier	Ensemble Technique	Correctly Classified Instances	Accuracy	Inaccurately Classified Instances	Inaccuracy	Confusion Matrix
Naive Bayes	Bagging	666	97.511	17	2.489	431 13 4 235
	Dagging	661	96.7789	22	3.2211	432 12 10 229
	Boosting	665	97.3646	18	2.6354	431 13 5 234
Hidden Naive Bayes	Bagging	653	95.6076	30	4.3924	435 9 21 218
	Dagging	633	92.6794	50	7.3206	437 7 43 196
	Boosting	647	94.7291	36	5.2709	433 11 25 214
BayesNet	Bagging	666	97.511	17	2.489	431 13 4 235
	Dagging	662	96.9253	21	3.0747	431 13 8 231
	Boosting	666	97.511	17	2.489	431 13 4 235

TABLE III ACCURACY OF STACKING AND STACKING-VOTING MODELS

Base Classifier	MetaClassifier	Accuracy	Correctly Classified Instances	Incorrectly Classified Instances	Inaccuracy	Confusion matrix
S-NB+LR	REPTREE	97.8038	668	15	2.1962	431 13 2 237
S- BayesNet+LR	REPTree	97.2182	664	17	2.7818	430 14 5 234
S-HNB+LR	REPTree	96.4861	659	24	3.5139	431 13 11 228
S- NB+ SMO	REPTree	97.8038	668	15	2.1962	431 13 2 237
S-BayesNet+ SMO	REPTree	97.2182	664	19	2.7818	430 14 5 234

S-HNB+ SMO	REPTree	96.9	662	21	3.0747	431 13 8 231
S-V- BayesNet+SGD	Voting- REPTree+Decisi on Stump	97.6574	667	16	2.3426	430 14 2 237
S-V- NB+SMO	Voting- REPTree+Rando m Forest	97.8038	668	15	2.1962	431 13 2 237
S-V- BayesNet+SMO	Voting- REPTree+Rando m Forest	97.2182	664	19	2.7818	430 14 5 234
S-V-- HNB+SMO	Voting- REPTree+Rando m Forest	96.7789	661	22	3.2211	429 15 7 232

B. Performance Measures Used

Performance measures of individual models are given in Table IV and that of bagging-dagging- boosting ensembles in Table V. The confusion matrix of each classifier is shown in the tables.

Other performance measures used are $\text{Precision} = \frac{TP}{TP+FP}$, $\text{Recall} = \frac{TP}{TP+FN}$, Equation (5)

F measure is a weighted harmonic mean of precision and recall. $F \text{ measure} = 2x \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$, Equation (6)

Matthews Correlation Coefficient is the geometric mean of chance corrected variants that describes the confusion matrix using a single digit, and lies between [-1,1], 1 describes a perfect condition, 0 a random one and -1 a poor condition.

$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$. Equation (7)

Accuracy is measured by the area under the ROC curve. An area of 1 denotes a perfect test The ROC area of all the models varied between 0.980-0.994, with bagging and boosting models showing slightly higher ROC area. Table VI gives the performance metrics of Stacking and Stacking-Voting models. In cancer detection high sensitivity is preferred. TPR is the sensitivity or recall. $FPR = \frac{FP}{TN+FP}$ Equation (8)

TABLE IV PERFORMANCE MEASURES OF INDIVIDUAL MODELS

Classifier	TPRate	FPRate	Precision	Recall	F measure	MCC	ROC Area	Time
Naive Bayes	0.974	0.024	0.974	0.974	0.974	0.943	0.994	0.01sec
Hidden Naive Bayes	0.960	0.058	0.961	0.960	0.960	0.913	0.990	0.01 sec
BayesNEt	0.977	0.018	0.977	0.977	0.977	0.949	0.993	0.03 secs

TABLE V PERFORMANCE MEASURES OF BAGGING- DAGGING- BOOSTING MODELS

Classifier	Ensemble Method	TPRate	FPRate	Precision	Recall	F measure	MCC	ROC Area	Time
Naive Bayes	Bagging	0.975	0.021	0.975	0.975	0.975	0.946	0.993	0.02 sec
	Dagging	0.968	0.037	0.968	0.968	0.968	0.929	0.994	0-01 sec
	Boosting	0.974	0.024	0.974	0.974	0.974	0.943	0.985	0.01 sec
Hidden Naive Bayes	Bagging	0.956	0.064	0.956	0.956	0.956	0.903	0.990	0.02 sec
	Dagging	0.927	0.122	0.930	0.927	0.925	0.839	0.992	0.01 sec
	Boosting	0.947	0.077	0.947	0.947	0.947	0.883	0.983	0.04 sec
BayesNet	Bagging	0.975	0.021	0.976	0.975	0.975	0.946	0.993	0.01 sec
	Dagging	0.969	0.032	0.970	0.969	0.969	0.933	0.994	0.01 sec
	Boosting	0.975	0.021	0.976	0.975	0.975	0.946	0.984	0.01 sec

TABLE VI PERFORMANCE METRICS OF STACKING AND STACKING - VOTING MODELS

Base Classifier	Meta Classifier	TP Rate	FP Rate	Precision	Recall	F measure	MCC	ROC Area	Time
S-NB + LR	REPTree	0.978	0.016	0.979	0.978	0.978	0.953	0.989	1.65 sec
S-HNB+LR	REPTree	0.965	0.040	0.965	0.965	0.965	0.923	0.967	1.63 sec
S-BayesNet+LR	REPTree	0.972	0.025	0.973	0.972	0.972	0.940	0.988	1.62 sec
S-NB+ SMO	REPTree	0.978	0.016	0.979	0.978	0.978	0.953	0.987	0.43 sec
S-BayesNet+ SMO	REPTree	0.972	0.025	0.973	0.972	0.972	0.940	0.988	0.44 sec
S-HNB+ SMO	REPTree	0.969	0.033	0.970	0.969	0.969	0.933	0.980	0.45 sec
S-V - BayesNet+SGD	Voting-REPTree+Decision Stump	0.977	0.016	0.977	0.977	0.977	0.950	0.988	0.91 sec
S-V- NB+SMO	Voting-REPTree+Random Forest	0.978	0.016	0.979	0.978	0.978	0.953	0.988	0.49 sec
S-V- BayesNet+SMO	Voting-REPTree+Random Forest	0.972	0.025	0.973	0.972	0.972	0.940	0.989	0.53 sec
S-V- HNB+SMO	Voting-REPTree+Random Forest	0.968	0.031	0.968	0.968	0.968	0.930	0.988	0.55 sec

C. Error Measures Used

The error measures used are Kappa Statistic, Mean absolute error, Root mean squared error, Relative absolute error. Table VII and Table VIII give the error metrics of the individual as well as that of the bagging-dagging-boosting models respectively. Table IX gives the error measures of the Stacking and Stacking-voting models. Kappa statistic is a measure that compares Observed Accuracy with Expected Accuracy. A perfect value is given by 1.

$$\text{Kappa} = \frac{(\text{observed accuracy} - \text{expected accuracy})}{(1 - \text{expected accuracy})}$$

Equation (9)

Mean Absolute error is calculated as the average of variance between the predicted and actual values. Root mean squared error measures the differences between values predicted by a model and the values observed. A perfect RMSE measure value is zero. Lower the RMSE, better is the prediction. Relative Absolute error measures the prediction accuracy of the classifier. A good model produces a value near to 0 and a bad model gives values greater than 1. If p_i is the predicted value and a_i the actual value for $i = 1, 2, 3, \dots, n$ instances of a model then RMSE, MAE and RAE, as given by Hill in 2012, are calculated as in equation 10

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (p_i - a_i)^2}{n}} \quad MAE = \frac{\sum_{i=1}^n |p_i - a_i|}{n} \quad RAE = \frac{\sum_{i=1}^n |p_i - a_i|}{\sum_{i=1}^n |\bar{a} - a_i|}$$

Equation (10)

Kappa statistic values is lowest for Hidden naive base classifier models. Naive Bayes and Bayes Network have better kappa values of 0.94. For the stacking ensembles Naive Bayes has a better kappa value of 0.9522. MAE values are worst for Hidden Naive bayes among the three Bayesian methods. Similiar are the cases with other error measures as shown in the tables VII-IX.

TABLE VII ERROR MEASURES OF INDIVIDUAL MODELS

Classifier	Kappa Statistic	Mean Absolute Error(MAE)	Root Mean Squared Error{RMSE}	Relative Absolute Error{RAE}	Total Relative Squared Error
Naive Bayes	0.9425	0.0257	0.1524	5.6471%	31.9596%
Hidden Naive Bayes	0.9122	0.0518	0.1823	11.3912%	38.2178%
BayesNet	0.949	0.0259	0.1505	5.6814%	31.5624%

TABLE VIII ERROR MEASURES OF BAGGING-DAGGING-BOOSTING MODELS

Classifier	Ensemble Method	Kappa Statistic	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Total Relative Squared Error
Naive Bayes	Bagging	0.9458	0.0253	0.1507	5.5575%	31.6062%
	Dagging	0.9293	0.0363	0.1605	7.9675%	33.657%
	Boosting	0.9425	0.0272	0.1523	5.9773%	31.9363%
Hidden Naive Bayes	Bagging	0.9023	0.0613	0.1883	13.4609%	39.4892%
	Dagging	0.8333	0.119	0.2351	26.1401%	49.2963%
	Boosting	0.8826	0.0506	0.2197	11.1127%	46.062%
BayesNet	Bagging	0.9458	0.0253	0.1507	5.5575%	31.6062%
	Dagging	0.9327	0.0326	0.1545	7.1534%	32.3859%
	Boosting	0.9458	0.0267	0.1516	5.8638%	3.7747%

TABLE IX ERROR MEASURES OF STACKING AND STACKING -VOTING MODELS

Base Classifier	Meta Classifier	Kappa Statistic	Mean Absolute Error	Root Mean Squared Error	Relative Absolute Error	Total Relative Squared Error
S- Naive Bayes+ LR	REPTree	0.9522	0.0383	0.1424	8.4212%	29.8486%
S- Hidden Naive Bayes+LR	REPTree	0.9229	0.0541	0.1746	11.8967%	36.6163%
S- BayesNet+LR	REPTree	0.9394	0.0421	0.151	9.2496%	31.6589%
S- NB+ SMO	REPTree	0.9522	0.0409	0.1447	8.9947%	30.3324%
S- BayesNet+ SMO	REPTree	0.9394	0.0445	0.1531	9.7681%	32.0996%
S- HNB+ SMO	REPTree	0.9327	0.0511	0.17	11.219%	35.6497%
S-V - BayesNet+SG D	Voting-REPTree+ Decision Stump	0.9491	0.044	0.1495	9.6678%	31.349%
S-V NB+SMO	Voting-REPTree+ Random Forest	0.9522	0.0388	0.148	8.5344%	31.0356%
S-V BayesNet+SMO	Voting-REPTree+ Random Forest	0.9394	0.042	0.1546	9.2394%	32.4052%
S-V HNB+SMO	Voting-REPTree+ Random Forest	0.9297	0.0511	0.1725	11.2189%	36.1714%

D. Comparison of the three Bayesian methods

In naive Bayes bagging and boosting were better than the individual method and the new stacking and stacking voting produced a better accuracy%. Stacking and Stacking Voting ensemble performance were similar in accuracy. With Bayesian networks bagging, dagging and boosting did not produce better results than the individual method. Stacking and Voting ensembles with Stochastic Gradient Descent produced the same outcome as with the individual classifier. Hidden Naive Bayes showed reasonable improvement while using stacking and stacking voting ensembles even though Bagging dagging and boosting ensembles decreased performance. The stacking Voting ensemble with SMO was seen effective for Hidden Naive Bayes classifier. Figure 6 shows a comparison of all the classifier performances and figures 3-5 give the accuracy performance of each classifier individually. F Score of all the models are illustrated in the figures 7- 9. F Score is worst in all cases of HNB except stacking with SMO model. F Score is best for the S-VNB+ SMO model. Dagging models also have the least F-Score. Kappa Statistic of all models is shown in Figures 10-12. They also show a similar trend as the F - Score values.

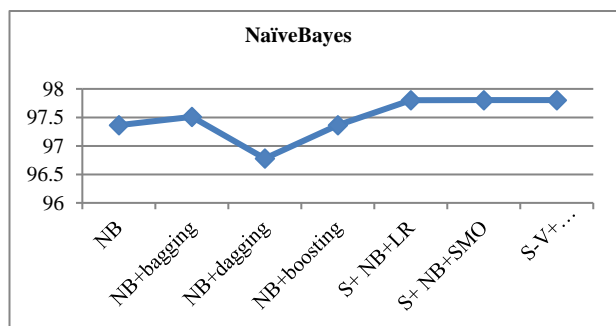


Fig.3 Comparison of Naive Bayes ensembles in accuracy

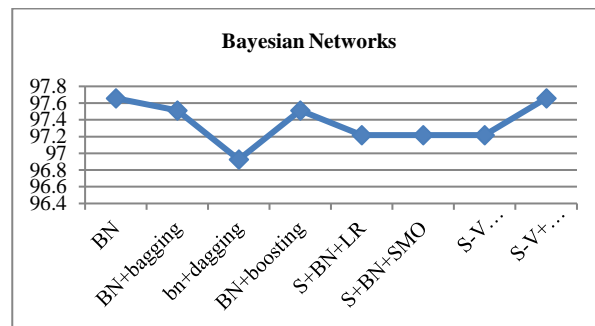


Fig. 4 Comparison of Bayesian Network ensembles in accuracy

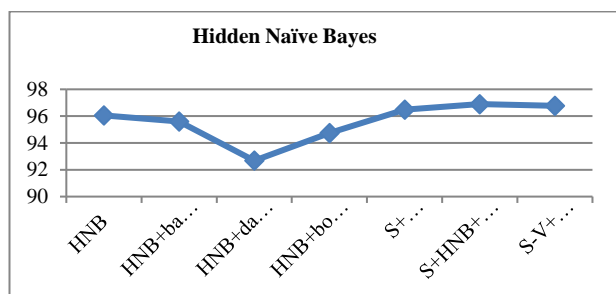


Fig. 5 Comparison of Hidden Naive Bayes ensembles in accuracy

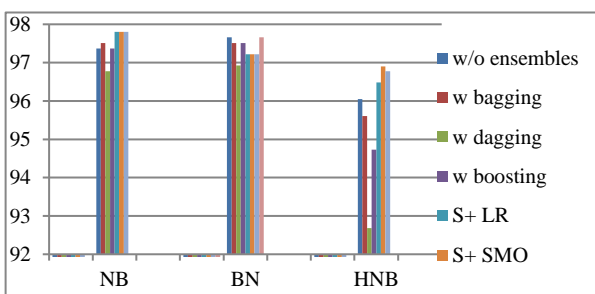


Fig. 6 Comparison of accuracy of Ensemble Performance

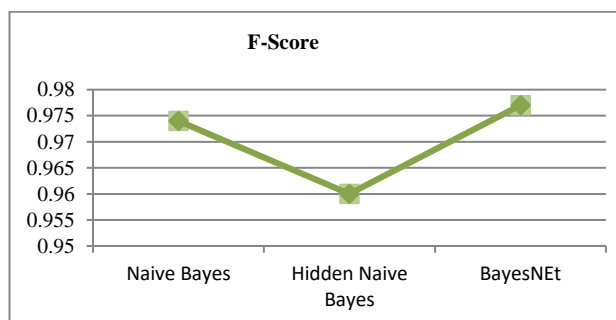


Fig.7 F- Score of Individual Bayesian Methods

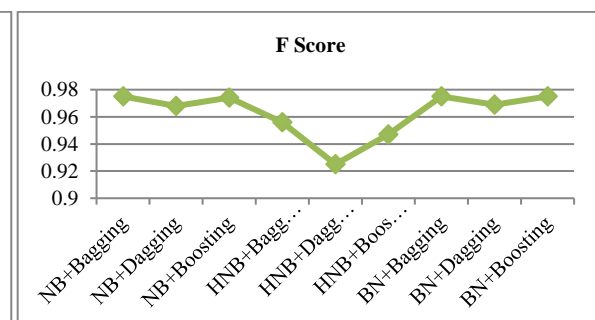


Fig.8 F- Score of Bagging-Dagging-Boosting Models

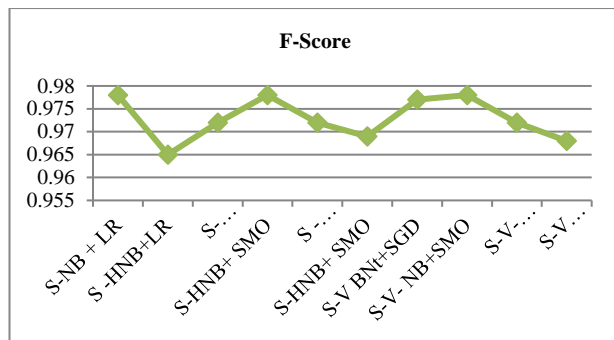


Fig. 9 F- Score of Stacking and S-V Models

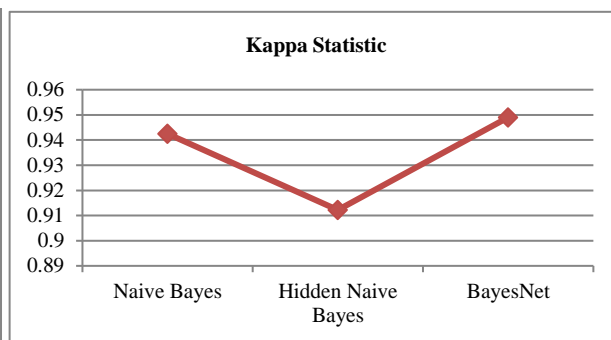


Fig. 10 - Kappa Statistic of Bayesian Models

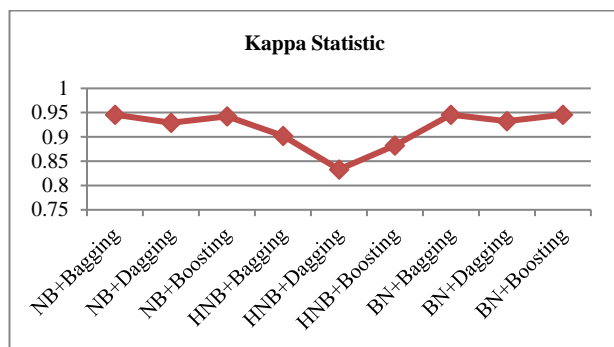


Fig. 11 Kappa Statistic of Bagging-Dagging-Boosting Models

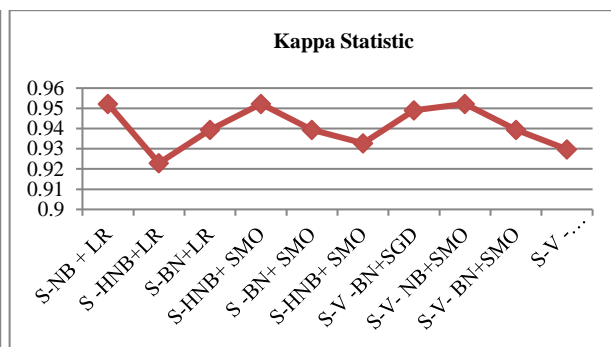


Fig. 12 Kappa Statistic of Stacking and S-V Models

V. CONCLUSION

The performance of ensemble of Bayesian methods are much more promising than individual Bayesian Methods. Naive Bayes ensembles were seen to give a prediction accuracy of 97.8%. The models need to be tested against other datasets. Additional Ensemble techniques such as multi boosting, rotation forests, random subspace, that partitions features instead of instances, and many more can be studied and analyzed for prediction performance on different datasets and tools. Besides, removal of irrelevant features[20], feature selection techniques[33] and optimization techniques[30] can be applied to improve the classifier performance. The key to strong ensemble models is using different models. Hence assorted combinations and multiplicity of heterogeneous classifiers as base and meta classifiers can be explored, so that the weakness of one classifier is overcome by the other classifier.

ACKNOWLEDGMENT

The authors acknowledge all those who provided valuable suggestions and advice. and also Dr William H Wolberg, Madison Hospitals, Wisconsin University for the dataset used.

REFERENCES

- [1] Vundavilli, H. , Datta,A. , Sima,C., Hua,J., Lopes,R., Michael Bittner, 2019, " Bayesian Inference Identifies Combination Therapeutic Targets in Breast Cancer", *IEEE Transactions On Biomedical Engineering*, Vol. 66, No. 9, September 2019
- [2] Witteveen, A. , Nane,G.F., Vliegen, I.M.H., Siesling,S. , IJzerman, M.J., 2018, "Comparison of Logistic Regression and Bayesian Networks for Risk Prediction of Breast Cancer Recurrence", *Medical Decision Making*, Volume 38, issue 7, 2018
- [3] Dong, X., Yu,Z., Cao,W., Shi, Y., Qianli M.A. ,2019, "A survey on ensemble learning", *Frontiers of Computer Science*, Volume 14, Issue 2, pp 241–258
- [4] Tribhuvan A.P., Tribhuvan P.P., Gade J.G. (2015) ,"Applying Naive Bayesian Classifier for Predicting Performance of a Student Using WEKA", *Advances in Computational Research*, ISSN: 0975-3273 & E-ISSN: 0975-9085, Volume 7, Issue 1, pp.-239-242.
- [5] K.M.Al-Aidaros, A.A. Bakar, Z.Othman, 2012, "Medical Data Classification with Naive Bayes Approach", *Information Technology Journal* 11, PP 1166- 1174
- [6] Nugroho,K.A., Setiawan,N.A., Adji,T.B.,(2013),"Cascade Generalization for Breast Cancer Detection" ,*IEEE*
- [7] Murat Karabatak, 2015,"A New classifier for breast cancer detection based on Naive Bayesian", *Measurement* 72, pp 32-36
- [8] Lopez,V.R., Barbosa, R.C., 2014, "On the Breast Mass Diagnosis Using Bayesian Networks", *Springer International Publishing*,2014,pp 474-485
- [9] I M D Maysanjay,I M A Pradnyana, I M Putrama, 2017,"Classification of breast cancer using Wrapper and Naïve Bayes algorithms", *International Conference on Mathematics and Natural Sciences (IConMNS 2017)* IOP Publishing,

- [10] Banu, B. A., Thirumalaikolundusubramanian, P., 2018, "Comparison of Bayes Classifiers for Breast Cancer Classification", *Asian pacific Journal of Cancer Prevention*, 2018; 19(10): 2917–2920.
- [11] Langarizadeh, M., Moghbeli, F., 2016, "Applying Naive Bayesian Networks to Disease Prediction: a Systematic Review", *Acta Informatica Medica*. 2016 OCT; 24(5): 364-369
- [12] Güzel C., Kaya M. & Yıldız O. Hasan Şakir Bilge, 2013, "Breast Cancer Diagnosis Based on Naïve Bayes Machine Learning Classifier with KNN Missing Data Imputation". *AWER Procedia Information Technology & Computer Science*. [Online]. 2013, 04, pp 401-407
- [13] Rath, M., Singh, A.K., 2012, "Breast Cancer Prediction using Naïve Bayes Classifier", *International Journal of Information Technology & Systems*, Vol. 1; No. 2: pp 77-80
- [14] Udayakumar E, Santhi S, Vetrivelan P, 2017, "An Investigation of Bayes Algorithm and Neural Networks for identifying the Breast Cancer", *Indian Journal of Medical Paediatric Oncology*, 2017, 38, 340 - 344
- [15] Sesen MB, Nicholson AE, Banares-Alcantara R, Kadir T, Brady M (2013) "Bayesian Networks for Clinical Decision Support in Lung Cancer Care". *PLoS ONE* 8(12):1-13 : e82349. <https://doi.org/10.1371/journal.pone.0082349>
- [16] Yeulkar, K., Sheikh, S., 2017, R Analysis of SEER, "Breast Cancer Dataset Using Naive Bayes and C4.5 Algorithm", *International Journal of Computer Science and Technology*, Vol. 8, Issue 4, 2017, pp 43-45
- [17] Nahar, J., Chen, Y.P., Ali, S., 2007, "Kernel-based Naive Bayes classifier for Breast Cancer Prediction", *Journal of Biological Systems*, Vol. 15, No. 1 (2007) 17–25
- [18] T S Umamaheshari, P. Sumathi, S Babu, 2018, "Pertaining To Comparison Of Algorithms For Breast Cancer Diagnosis", *International Journal of Scientific Research and Innovations*, 5, 2, 2018, pp 28-35
- [19] Demigha, S., 2016, Mining Knowledge of the Patient Record: "The Bayesian Classification to Predict and Detect Anomalies in Breast Cancer", *The Electronic Journal of Knowledge Management* Volume 14 Issue 3 (pp128-139)
- [20] Shih, A., Arthur Choi, A., Darwiche, A., 2018, "Formal Verification of Bayesian Network Classifiers", *Proceedings of Machine Learning Research* vol 72, 427-438, 2018
- [21] Chang, M., Dalpatadu, R.J., Phanord, D., Singh, A.K., "Breast Cancer Prediction Using Bayesian Logistic Regression", *Open Access Biostatistics and Bioinformatics*, 2018, Volume 2, issue 3, pp 1-5 (Michael C, Rohan J D, Dieudonne P, Ashok K S. "Breast Cancer Prediction Using Bayesian Logistic Regression". *Open Acc Biostat Bioinform*. 2(3).)
- [22] Soria, D., Biganzoli, E., Ellis, I.O., Garibaldi, J.M., "A Comparison of Three Different Methods for Classification of Breast Cancer Data", 2008 Seventh International Conference on Machine Learning and Applications, IEEE
- [23] Le, T., 2018, "On the Interpretation of Ensemble Classifiers in Terms of Bayes Classifiers", *Journal of Classification* (2018) 35, pp 198-229
- [24] Joshi, J., Doshi, R., Patel, J., 2014, "Diagnosis And Prognosis Breast Cancer Using Classification Rules", *International Journal of Engineering Research and General Science* Volume 2, Issue 6, October-November, 2014, pp 315-323
- [25] S. Kharya, S. Agrawal, S. Soni, "Using Bayesian Belief Networks for Prognosis & Diagnosis of Breast Cancer", *International Journal of Advanced Research in Computer and Communication Engineering* Vol. 3, Issue 2, February 2014, pp 5423=27
- [26] Abdara, M., Moghadamb, M.Z., Zhouc, X., Gururajanc, R., Taod, X., Baruae, P.D., Gururajanf, R., 2018, "A new nested ensemble technique for automated diagnosis of breast cancer", *Pattern Recognition Letters*, 2018
- [27] Wan Nor Liyana Wan Hassan Ibeni, Mohd Zaki Mohd Salikon, Aida Mustapha, Saiful Adli Daud, Mohd Najib Mohd Salleh, "Comparative analysis on bayesian classification for breast cancer problem", *Bulletin of Electrical Engineering and Informatics* Vol. 8, No. 4, December 2019, pp. 1303-1311
- [28] Zhang, H., Jiang, L., Su, J., 2005, *Hidden Naive Bayes*, AAAI-05 /919- 924
- [29] Ratnawati, D.E., Priandani, N.D., Machsus, "A Modified K-Means with Naïve Bayes (KMNB) Algorithm for Breast Cancer Classification", *Journal of Telecommunication, Electronic and Computer Engineering*, Vol. 10 No. 1-6, pp 137-140
- [30] Beretta, S. Castelli, M., Goncalves, I., Merelli, I., Ramazzotti, D., 2017, "Combining Bayesian Approaches and Evolutionary Techniques for the Inference of Breast Cancer". *Networks, arXiv.org*
- [31] Kotsianti S.B., Kanellopoulos D. (2007), "Combining Bagging, Boosting and Dagging for Classification Problems". In: Apolloni B., Howlett R.J., Jain L. (eds) *Knowledge-Based Intelligent Information and Engineering Systems*. KES 2007. Lecture Notes in Computer Science, vol 4693. Springer, Berlin, Heidelberg
- [32] Jian-Fang Chang, Na Dong, Wai Hung Ip, Kai Leung Yung, 2019, "An ensemble learning model based on Bayesian model combination for solar energy prediction", *J. Renewable Sustainable Energy* 11, 043702 (2019); pp 1-14
- [33] Mathew, T.E. (2019). "A Logistic Regression with Recursive Feature Elimination Model for Breast Cancer Diagnosis". *International Journal on Emerging Technologies*, 10(3): 55–63
- [34] Yaguang Ji, Songnian Yu, Yafeng Zhang, 2011, "A novel Naive Bayes model: Packaged Hidden Naive Bayes", *IEEE*, pp 484-487
- [35] Ganpati, A., 2016, "A Performance Comparison Of End, Bagging And Dagging Meta Classification Algorithms", *International Journal of Advances in Electronics and Computer Science*, Volume-3, Issue-4, Apr.-2016, 81-83
- [36] Kathija, Nisha, S., 2016, "Breast Cancer Data Classification Using SVM and Naïve Bayes Techniques", *International Journal of Innovative Research in Computer and Communication Engineering*, Vol. 4, Issue 12, December 2016, pp 21167-21172

Simple And Ensemble Decision Tree Classifier Based Detection Of Breast Cancer

Tina Elizabeth Mathew

Abstract: Breast cancer being the top cancer, among all other types, in women worldwide has an increasing incidence particularly in developing countries where the majority of cases are diagnosed in late stages. Low survival is attributed to mainly late diagnosis of the disease, which is ascribed to lack of early diagnostic facilities. Many techniques are being used to aid early diagnosis. Besides medical and imaging methods, statistical and data mining techniques are being implemented for breast cancer diagnosis. Data mining techniques provide promising results in prediction and employing these methods can help the medical practitioners in quicker disease diagnosis. Numerous supervised techniques are being deployed to improve prediction and diagnosis. Decision trees are supervised learning models which provide high accuracy, stability and ease of interpretation. In this study, different Decision tree models, single and ensemble methods, are implemented and their performance is evaluated using the Wisconsin breast cancer diagnostic data. Rotation Forest classifier was seen to produce the best accuracy while using ensembles and NBTree in single models for disease diagnosis.

Keyword: Decision tree(DT), Naive Bayes Tree(NB Tree), REPTree, Rotation Forest, Bagging, Boosting, Ensembles, Random Forest, Adaptive Boosting(AdaBoost)

1.INTRODUCTION

Decision trees are considered to be among the topmost few methods that aid in efficient supervised classification. They build classification models as a treelike structure with internal nodes also known as chance nodes representing attributes, branches showing outcomes and leaf nodes or end nodes giving the class labels. The first main node called root or decision node represents the whole sample. The path from root to leaf gives the classification rule. A decision tree creates a training model which predicts class or value of target variables by decision rules that are inferred from the training data provided. The first challenge is to select the correct attribute as the root node and internal nodes. Another challenge is overfitting. To find the attribute that is to become the root node the many attribute selectors can be used one being information gain measure. In Information gain the attribute with highest information gain is chosen as root. The steps involved in building a tree are splitting, stopping, and pruning. The nodes are identified using characteristics such as Gini Index, Gini Ratio or Information gain beginning from the root node. Splitting is done at the root node and subsequent internal nodes are created and is continued till a stopping criteria is met. To prevent overfitting Stopping rules are defined, some being 1) the minimum number of records in a leaf; (2) the minimum number of records in a node prior to splitting; and (3) the depth any leaf from the root node[1]. In cases where stopping rules are inadequate, Pruning is applied. Large decision trees are built and are then pruned to an optimal size by removing branches and nodes that give less information.

K- fold Cross Validation, using a Validation dataset, considering the proportion of records with error prediction are few ways to prune a tree. In this study various decision tree algorithms- J48, RepTree, CART, Hoeffding Trees, BFTree, Functional Tree, LMT, Decision Stump, NBTree, ADTree, Random Forest, Random Tree, ADaBoost, and Rotation Forest are evaluated for their classification accuracy of breast cancer using the WEKA Environment. Single decision trees and ensemble methods are discussed and their performance on the breast cancer dataset is evaluated.

1.1 RELATED WORK

Song[1] et al studied various decision tree algorithms including CART, C4.5, CHAID, and QUEST using SAS and SPSS and analyzed the effectiveness of decision trees. Al-Sahily[2] et al in their study used different decision tree algorithms J48, Function Tree, Random Forest Tree, AD Alternating Decision Tree, Decision stump and Best First and compared their performance in weka. It was observed that highest precision 97.7% was with Function Tree classifier with highest correct number of instances but, a precision of 88% was found in Decision stump with lowest correct number of instances. Yadav[3] et al in their work compared decision tree and Artificial Neural Network breast cancer data. The results showed that algorithms produced an overall prediction accuracy for decision tree from 90% to 94%, and for SVM from 94.5% to 97%. Bhargava[4] et al studied about Decision Trees with the Univariate and the Multivariate approaches. It was observed that Multivariate Decision Tree used the concept of attributes correlation and provided a better way to perform conditional tests when compared to Univariate approach. Kaur[5] et al in their work used the data mining tool WEKA as an API of MATLAB for generating modified J-48 classifiers. The experimental results showed an improvement over the existing J-48 algorithm by providing an accuracy up to 99.87 %. Rai[6] et al in their paper proposed the concept of correlation with Best Fit Search(CFS) in attribute selection to find the relevant factors that affected dropping out of students from B.Tech

- Research Scholar
 - Faculty of Applied Science and Technology
 - University of Kerala, Thiruvananthapuram
 - Kerala, India Email: tinamathew04@gmail.com
- IJSTR©2019

and BCA courses using ID3 algorithm. Out of the available 33 attributes 12 were selected using CFS. An accuracy of 98.18 was obtained. Pritom[7] et al in their work used 3 classifiers Naive Bayes, C4.5 Decision Tree and Support Vector Machine to predict breast cancer recurrence. They ranked the features using the rankers algorithm and reduced the attribute set based on the ranking to enhance accuracy of prediction. In the experimental results SVM outperformed all other classifiers. It was seen that feature selection improved prediction accuracy of Decision trees. Nalini[8] et al compared Naive Bayes and J48 for prediction accuracy in breast cancer diagnosis. Significant features were selected by applying the Consistency Subset Evaluator. Naive Bayes was found a better classifier. Khadhim[9] in his work used Id3 classifier tree algorithm to classify the breast cancer risk factors and determined which factor was more effective than others and also studied the odds of developing breast cancer for those having a genetic mutation with family history. Cha[10] et al used Genetic algorithms to select best attributes as well as to construct short, and near-optimal decision trees. Decision trees were transformed into chromosomes, for this a new scheme to encode and decode a decision tree to and from a chromosome was proposed. Alickovic[11] et al in their work applied Genetic Algorithm based feature selection on various classifiers Logistic Regression, Decision Trees, Random Forest, Bayesian Network, MLP, RBFN, SVM and Rotation Forest, to predict breast cancer. It was seen that Rotation Forest model with GA-based 14 features gave the best prediction accuracy of 99.48. It was concluded that ensemble of more simple classifiers gave good results than complex methods. Wang [12] et al in their paper proposed a new decision tree classifier Self adaptive NBTree which is a hybrid of Naive Bayes and Decision trees. The Bayes measure helps in resolving overgeneralization and overspecialization. It was found better than NBTree and C4.5. Fruend[13] et al introduced ADTrees and analyzed the performance of the classifier. ADTree is similar to C5.0 with boosting and produces the same error rate as it. Venketasan [14] et al studied four decision tree algorithms J48, CART, ADTree and BFTree. J48 was seen to show a high accuracy of 99% followed by BFtree with 98% ADTree with 97% and CART with 96%. Snousy[15] et al in their study applied single and ensemble methods. on gene expression data. They used chisquare and gain ratio attribute selection methods to select features. Ensemble methods were found to increase accuracy. In bagging C4.5 and REPTrees were used and in boosting AdaBoost, Random Forests and ADTree were used. Juneja [16] et al in their work proposed a feature rank based improved decision tree and applied it on two breast cancer datasets. Chi square test was applied on the attributes and they were ranked. These ranked attributes were used to produce a decision tree. This was compared against the decision

tree, naive bayes, random tree and random forest classifiers and was found to predict breast cancer accurately. Mishra[17] et al in their study compared Random Trees and Random Forest classifiers with attribute selector filter for feature extraction on microarray datasets and found that attribute selection filter based Random Tree classification selection produces better classification accuracy than that of Random Forest. Rodriguez[18] et al proposed the Rotation forest ensemble and found that it outperformed the standard implementations of Bagging, AdaBoost, and Random Forest available in WEKA. 33 data sets from UCI Machine Learning Repository was used and they showed that Rotation Forest outperformed all three methods by a large margin. Banu[19] in her work compared J48 and Decision Stump algorithms and came to the conclusion that the J48 model was much better than the Decision stump model. Arundathi[20] et al in their paper studied Six decision tree classifiers -Hoeffding, REP Tree, Decision Stump, Random Tree, Random Forest and J48 on an educational dataset and found that random tree, Random Forest and J48 algorithms showed the best performance. Bindhia[21] et al in their study compared various decision tree algorithms and concluded that NB tree is the best suited for decision making as it gave 75.3% of correctly classified instances in the data set used. Sharma [22] et al in their work compared the various decision tree algorithms in weka against 5 data sets in UCI and KEEL and found that many decision tree models like LMT and J48 were good performers. Sumbaly[24] et al in paper evaluated the working of J48 decision tree and observed an accuracy of 94.5637 in predicting Breast cancer.

2. MATERIALS AND METHODS

2.1 DATA SET

The data used is the Original Wisconsin Breast Cancer Database(WBCD) obtained from Dr. William H. Wolberg of Wisconsin University Hospitals, Madison. The Original data set has 699 instances each with 11 attributes. The ID of an instance, the first attribute is discarded as it has no role in prediction, different characteristics of an instance is given in the remaining 9 attributes. These attributes are the cytological characteristics of the breast fine needle aspiration(FNA) test. They all have a value between 1 and 10 with increasing intensity from 1 to 10 depending on the values of the attribute. The last attribute Class shows the diagnosis made. Each instance belongs to either one of the 2 possible classes, benign with value 2 or malignant with value 4. The database has 699 instances with 458 benign cases - 65.5% and 241 malignant cases- 34.5%. Sixteen instances were avoided due to missing values and 683 instances were taken for the study, of which 444 belong to benign class and 239 to malignant class. Table 1 shows the attributes used in the dataset.

Table 1 Attributes of the WBC Dataset

Attributes of the data used	Clump Thickness	Uniformity of Cell Size	Uniformity of Cell Shape	Marginal Adhesion	Single Epithelial Cell Size
	Bare Nuclei	Bland Chromatin	Normal Nucleoli	Mitoses	

2.2 WEKA

The open source software Waikato Environment for knowledge Analysis 3.8.3(WEKA) developed by the University of Waikato, New Zealand is used for evaluating the simple and ensemble decision tree classifiers. Weka contains a collection of data mining tools which can be used for various purposes like preprocessing classification, clustering, regression, rule mining and many more tasks. It provides many feature selection and data conversion methods. It is written in Java and the functions can be used directly on any provided dataset or embedded in our code. The dataset used in weka is in the format arff (Attribute-Relation File Format). The main interface of Weka is the Explorer. It provides many panels helpful in various data mining tasks. Besides this the Experimenter helps in predictive performance analysis of Weka's machine learning algorithms on collection of datasets.

2.3 METHODOLOGY USED

2.3.1 Decision Trees

Decision trees are of various types ID3(Iterative Dichotomiser 3), C4.5, C5.0, M5 model tree, Decision Stump, CART(Classification And Regression Tree), CHAID(Chi-squared Automatic Interaction Detector), MARS(Multivariate Adaptive Regression Splines), Conditional Inference Trees, CRUISE(Classification Rule with Unbiased Interaction Selection and Estimation),GUIDE(Generalized, Unbiased, Interaction Detection and Estimation),QUEST (Quick, Unbiased, Efficient Statistical Tree), LOTUS(Logistic Regression Tree with Unbiased Selection) and many more. Besides these decision trees can be used to create ensemble methods which have more than one decision tree such as Bagging decision trees, ADTree, Random Forest, Boosting decision tree, AdaBoost, Rotation forest and Decision Lists. Bagging or Bootstrap Aggregating models help to decrease the variance of the model and boosting models decrease model bias. Learning capacity of models are impaired by variance noise and bias. Bagging suffers from bias and boosting from overfitting. Ensemble methods help in increasing stability of models by decreasing error. In order to build decision trees a root node is to be selected. The feature which is to be the root from among the 9 attributes of the breast cancer data set is selected based on various measures such as information gain, gain ratio, correlation, and chisquare. Filters used in Waikato Environment for Knowledge Analysis (Weka) and their comparison is shown in table 1. The performance of 13 decision tree models are evaluated using the WBC dataset.

Table 2 Feature selection filters

Infogainattribute evaluator	Correlation	Gainratio	ChiSquare	OneR attribute evaluator	Symmetrical ranking filter	ReliefF Ranking filter
.702 size_uniformity	2 0.61 bare_nucleoli	6 0.303 bare_nucleoli	6 539.79308 size_uniformity	2 92.5329 shape_uniformity	3 0.429 size_uniformity	2 .6012 bare_nucleoli
0.677 shape_uniformity	3 0.535 size_uniformity	2 0.3 size_uniformity	2 523.07097 shape_uniformity	3 91.8009 size_uniformity	2 0.412 bare_nucleoli	6 0.5426 shape_uniformity
0.603 bare_nucleoli	6 0.53 normal_nucleoli	8 0.272 shape_uniformity	3 489.00953 bare_nucleoli	6 90.776 bland_chromatin	7 0.395 shape_uniformity	2 0.5372 size_uniformity
0.555 bland_chromatin	7 0.483 epithelial_size	5 0.237 normal_nucleoli	8 453.20971 bland_chromatin	7 90.0439 epithelial_size	5 0.331 epithelial_size	1 0.4734 clump_thickness
0.534 epithelial_size	5 0.482 shape_uniformity	3 0.233 epithelial_size	5 447.86118 epithelial_size	6 89.8975 bare_nucleoli	8 0.326 normal_nucleoli	7 0.4268 bland_chromatin
0.487 normal_nucleoli	8 0.468 9 mitoses	4 0.21 marginal_adhesion	8 416.63061 normal_nucleoli	8 89.7511 normal_nucleoli	7 0.3 bland_chromatin	5 0.2959 epithelial_size
0.464 marginal_adhesion	4 0.46 marginal_adhesion	7 0.201 bland_chromatin	4 390.0595 marginal_adhesion	1 85.9444 clump_thickness	4 0.295 marginal_adhesion	8 0.2824 normal_nucleoli
0.464 clump_thickness	1 0.292 bland_chromatin	7 0.188 9 mitoses	1 378.08158 clump_thickness	4 85.9444 marginal_adhesion	1 0.233 clump_thickness	4 0.2439 marginal_adhesion
0.212 9 mitoses	1 0.214 clump_thickness	1 0.152 clump_thickness	9 191.9682 mitoses	9 78.7701 mitoses	9 0.205 9 mitoses	9 0.069 9 mitoses

1. C4.5

C4.5 proposed by Ross Quinlan, a modified version of the ID3 (Iterative Dichotomiser 3) algorithm, creates a decision

A. Decision tree Classifiers

tree based on the best predictive attribute at each node[15]. Gain ratio is used to find the best attribute for splitting. It handles missing data well and can be used for continuous and discrete values. A modified version of it is the C5.0 algorithm which supports boosting.

2. J48

J48 algorithm is the weka implementation of the C4.5 algorithm. J48 works taking into consideration the gain ratio. It gives the importance of the attribute and shows the amount of data it holds. Pruning is done using a confidence factor. Smaller the confidence factor more is the pruning. J48 effectively handles missing values in training data and continuous and discrete attributes.

3. Simple Cart

Classification and Regression Trees (CART) was introduced by Leo Breiman, Jerome Friedman, Richard Olshen and Charles Stone in 1984. It provides a basis to bagged decision trees and random forest. It is considered as a binary decision tree. It begins from a root node which is split into two daughter nodes. It supports numerical target values.

4. REPTree

Reduced Error Pruning Tree [REPTree] is a fast decision learner tree based on C4.5. It uses information gain / variance for building the decision tree. Information gain, known as Kullback-Leibler divergence, denoted by $IG(S,A)$ for a set S is the effective change in entropy after deciding on a particular attribute A . It measures the relative change in entropy with respect to the independent variables. The equation gives the information gain

$$IG(S,A) = H(S) - H(S,A) \quad (1)$$

or

$$IG(S,A) = H(S) - \sum p(x) \log_2 \left(\frac{1}{p(x)} \right) \quad (2)$$

5. Hoeffding Tree

Hoeffding Tree is an incremental decision tree algorithm produced by Domingos and Hulten in 2000 using the Hoeffding bound. It works on and learns from massive data streams on the assumption that its distribution does not change over time.

6. BFTree

Best First decision tree classifier first introduced by Haijian Shi in 2007 builds a tree by finding the best node and expanding it first. The best node is the node that maximally reduces impurity among all nodes available for splitting, i.e. non leaf nodes. A node is selected as the root and branches are created based on the Gini Index. then the dataset is partitioned into subsets and each branch - root is assigned a data subset. This step is repeated for each branch selecting the best subset among the subsets. This goes on till all nodes are pure i.e. all instances have the same class label or it reaches a specific number of expansions.

7. NBTTree

A hybrid of naive bayes and decision trees which uses the Bayes measure to construct decision trees. with naive bayes classifier at the leaves. It effectively handles over specialization and over generalization problems associated with decision trees[12]. It was proposed by Ron Kohavi in 1996.

8. Random Tree

Random Tree is a classifier that constructs a decision tree using k randomly chosen attributes. It does no pruning and class probabilities can be estimated. It is considered to be a weak classifier and when used in ensembles can be made stronger. A group of random trees form a random forest. It was introduced by Leo Breiman and Adele Cutler.

9. LMT

Logistic Model Trees or LMT build decision trees with logistic regression functions at the trees.

10. Decision Stump

It is a one level decision tree with the root node connected to the leaf nodes. They are called 1-rules because they predict based on the value of a single attribute. They are weak learners and are used as base learners for ensemble methods. It was introduced by Wayne Iba and Pat Langley in 1992.

11. Functional Tree

It is a generalization of multivariate trees[24]. A tree is built and then pruned. It uses the divide and conquer strategy to build the tree. The attribute with maximum gain ratio is chosen.

B. Ensembles

Ensemble classifiers are classified into Bagging, Boosting, Voting and Stacking. Ensemble classifiers create new classifiers that perform better than their constituent classifiers and reduce bias and variance.

1. Bagging

In bagging the data set of size n is partitioned into $k < n$ sample sets and the model is run in parallel on these sample sets. Sampling is done with replacement and data values can overlap. One of the sample is used as the test data and the rest are used as training data. The final result is obtained by combining the predictions given by the models based on their votes.

2. Random Forest

Random Forest first proposed by Tin Kam Ho in 1995, is an ensemble of decision trees following the divide and conquer approach. It creates many decision trees during training on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It is effective against the overfitting problem associated with decision trees and offers feature selection using gini index. The whole dataset is split into a random number of subsets and individual decision trees are generated for each newly created data subset. The Individual decision trees are generated based on gini index. The collection of trees are known as a forest. The outcomes from each tree is evaluated and the most voted/predicted value is taken as the final prediction.

C. Boosting

Boosting is an ensemble method developed by Freund. It assigns weights to the data items. Initially all data items have equal weights. Models are assigned weights too. Boosting works iteratively. For each new model created the wrongly classified instances of the previous model are used after assigning higher weights to them. The final model is thus derived using voting.

Some examples of boosting algorithms are AdaBoost, Gradient Boosting Method (GBM), XGBM, Light GBM, and CatBoost.

1. AdaBoost

AdaBoost or Adaptive Boosting, is the first practical boosting algorithm proposed by Freund and Schapire in 1996. It helps in making weak classifiers strong.

2. ADTree

Alternating Decision Trees(AD Trees) are decisions tree using boosting that generate easier and simpler rules for interpretation. It uses a measure of confidence known as classification margin and is used for binary classification problems. It uses two decision nodes a prediction node and a splitter node.

3. Rotation Forest

It is a tree based ensemble classifier developed by Juan J Rodriguez that uses feature extraction. It does transforms on the attributes and converts them into Principal Components. Rotation forest models generate less number of trees but with better performance when compared to Random Forest. An unsupervised technique Principal Component Analysis is applied for transformation of attributes. Here tree size used is 10.

D. Stacking

Stacking is a method in which a single training dataset is given to different models and trained. The training set is divided using k-fold validation and the resultant model is formed

3.1 Simple Decision Trees

The decision trees are built using various classifiers and 10 fold cross validation is done. Models for thirteen types of simple decision trees are built using the Weka tool. A comparison of the classifiers is shown in the table3 using various performance measurement parameters. It was observed that the best accuracy of 97.07% was given by Naive Bayes Tree and the least accuracy, of 87.55%, was given by Decision Stump classifier. Performance and error measuring parameters are used like Kappa Statistic, Mean absolute error, Root mean squared error as shown in the table 3. Confusion matrix obtained for each classifier is also shown in the table. The best Cohen's Kappa coefficient was given by NB Tree with a value of 0.936. Kappa statistic compares the observed accuracy of the classifiers with the expected accuracy and is more precise in measuring accuracy. A value over 0.75 is excellent, and fair to good for values between 0.40 and 0.75 and poor for values below 0.40. The confusion matrix of each classifier on evaluation shows that the Random Forest classifier correctly classified 433 benign cases and 227 malignant cases. Naive Bayes Tree Classifier correctly classified 431 benign cases and 232 malignant cases. Figure1 gives the accuracy of prediction in percentage.

3. RESULTS AND DISCUSSION

Table 3 Performance of Simple Decision Tree Classifiers

Evaluation Criteria	Classifiers												
	J48	Hoefding Tree	NB Tree	REP Tree	Simple Cart	BF Tree	LMT	AD Tree	Random Forest	Random Tree	SYS For	Decision Stump	Functional Tree
Correct Instances	638	638	663	648	649	652	654	651	660	651	662	598	653
Incorrect instances	45	45	20	35	34	31	29	32	23	32	21	85	30
Accuracy Percentage	93.4	93.4	97.07	94.88	95.022	95.46	95.74	95.31	96.6325	95.31	96.92	87.55	95.6
Inaccuracy Percentage	6.59	6.59	2.9	5.12	4.978	4.53	4.24	4.68	3.36	4.68	3.07	12.44	4.39
Kappa statistic (KS)	0.855	0.855	0.936	0.8868	0.8912	0.8998	0.9066	0.8964	0.9259	0.8966	0.9329	0.7435	0.9035
Mean absolute error (MAE)	0.0843	0.0843	0.0295	0.0762	0.0759	0.0595	0.0561	0.0862	0.0676	0.0484	0.0841	0.1849	0.0483
Root mean squared error (RMSE)	0.2288	0.2288	0.1551	0.2014	0.2166	0.2105	0.1827	0.1959	0.166	0.1988	0.1722	0.311	0.1988
Relative absolute error (RAE)	18.5134	18.5134	6.4833	16.73	16.6768	13.0813	12.325	18.946	14.8517	10.6411	18.4853	40.6271	10.6116
Root relative squared error (RRSE)	47.9727	47.9727	32.5133	42.22	45.4112	44.12	38.2991	41.0783	34.8098	41.6799	36.105	65.215	41.6726

Confusi on Matrix	422 22 23 216	422 22 23 216	431 13 7 232	429 15 20 219	424 20 14 225	431 13 18 221	430 14 15 224	431 13 19 220	433 11 12 227	430 14 18 221	430 14 7 232	367 77 8 231	429 15 15 224
-------------------------	------------------------	------------------	-----------------------	------------------------	------------------	------------------------	------------------------	------------------------	------------------------	------------------------	-----------------------	--------------------	------------------

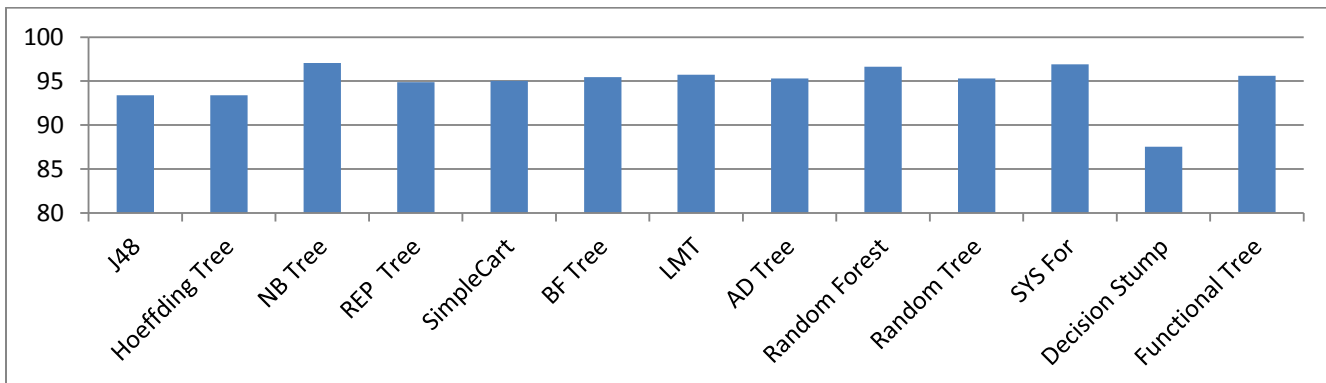


Figure 1 Simple decision tree classifiers vs Accuracy %

plotted in the figure. NBTree classifier produced the least error in prediction. Misclassification is highest in the Decision Stump classifier.

The misclassification rate is shown in figure 2. The inaccuracy percentage in prediction of all the classifiers are

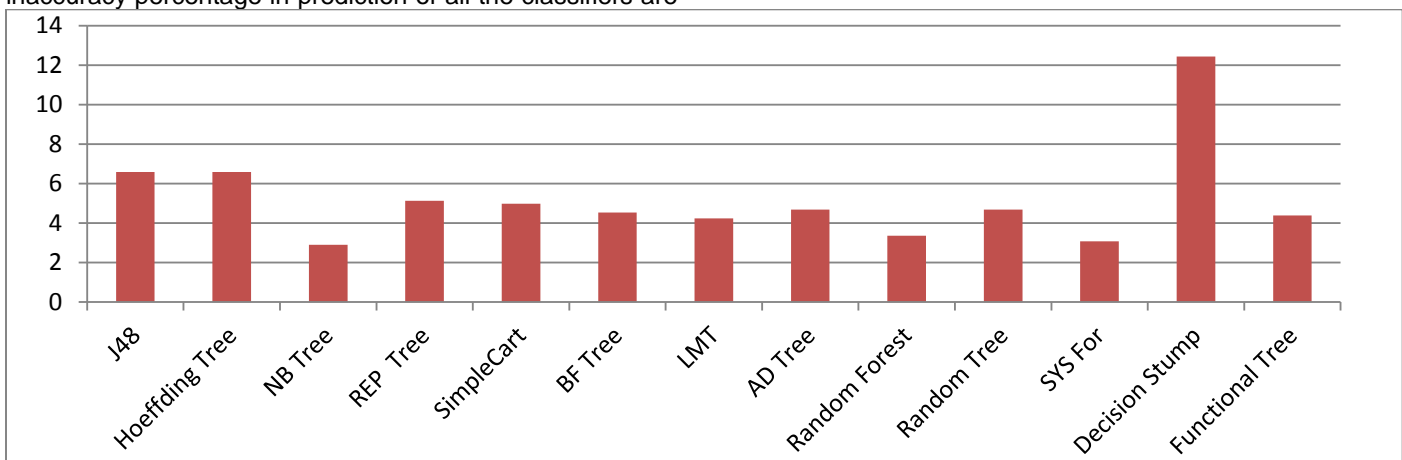


Figure 2 Decision tree Classifiers vs Inaccuracy %

3.2 Ensemble classifiers

3.2.1 Boosting with AdaBoost

The AdaBoost classifier was also run with 13 different base classifiers and the accuracy received for each classifier is shown in table. The accuracy percentage received with AdaBoost with base classifier Hoeffding Tree and SysFor classifiers produced the best accuracy prediction of 96.9%. , with Hoeffding Tree being built with lesser time than SysFor in 0.03 sec against 0.43. This is followed by the J48 classifier with 96.7 % accuracy in predictions. Performance measures used are represented in table 4. Kappa Statistic, Precision, Recall, F- Score, Matthews

Correlation Coefficient, ROC area and the Confusion Matrix is shown.

Time taken by each classifier is shown in table 4. Random Forest and J48 take the least time while LMT and NBTrees take longer time to build the model.

ROC area is a better illustration of precision. It plots True Positive Rates(TPR) against False Positive Rates (FPR). Hoeffding trees and SysFor have same accuracy of 96.9% but ROC area of SysFor, 0.982 is better than that of Hoeffding Trees at 0.978.

Confusion Matrix helps in evaluating the classifier by representing actual class against predicted class. Figure 3 gives the accuracy in classification by the classifiers

Table 4 Performance Analysis of AdaBoost

Base classifiers	Accuracy%	Time to build the model	Kappa statistic	Matthews Co relation Coefficient	Precision	Recall	F-measure	ROC Area	Confusion Matrix
Decision Stump	95.9	0.03 sec	0.9095	0.910	0.959	0.959	0.959	0.989	432 12 16 223
J48	96.77	0.01 sec	0.9293	0.929	0.968	0.968	0.968	0.990	432 12

									10 229
Hoeffding Tree	96.9	0.03 sec	0.9327	0.933	0.970	0/969	0.969	0.978	431 13 8 231
Base Classifiers used	Accuracy % using 10 CV	Time to build the model	Kappa Statistic	Matthews Co relation Coefficient	Precision	Recall	F-measure	ROC Area	Confusion Matrix

NB Tree	96.33	8.41sec	0.9199	0.920	0.964	0.963	0.963	0.986	429 15 10 229
REPTree	96.04	0.04ec	0.9132	0.913	0.961	0.960	0.960	0.984	430 14 13 226
Simple cart	95.31	1.39 sec	0.8968	0.897	0.953	0.953	0.953	0.987	429 15 17 222
SysFor	96.9	0.43 sec	0.9327	0.933	0.970	0.969	0.969	0.982	431 13 8 231
Random Tree	93.7	0 sec	0.8607	0.861	0.937	0.937	0.937	0.960	426 18 25 214
RandomForest	96.63	0.05 ec	0.9259	0.926	0.966	0.966	0.966	0.992	433 11 12 227
ADtree	96.48	0.1 sec	0.9228	0.923	0.965	0.965	0.965	0.988	432 12 12 227
BFTree	96.04	1.5 sec	0.9123	0.913	0.960	0.960	0.960	0.990	435 9 18 221
LMT	96.19	9.79sec	0.9163	0.916	0.962	0.962	0.962	0.981	431 13 13 226
Functional Tree	96.33	1.01 sec	0.9192	0.919	0.963	0.963	0.963	0.985	434 10 15 224

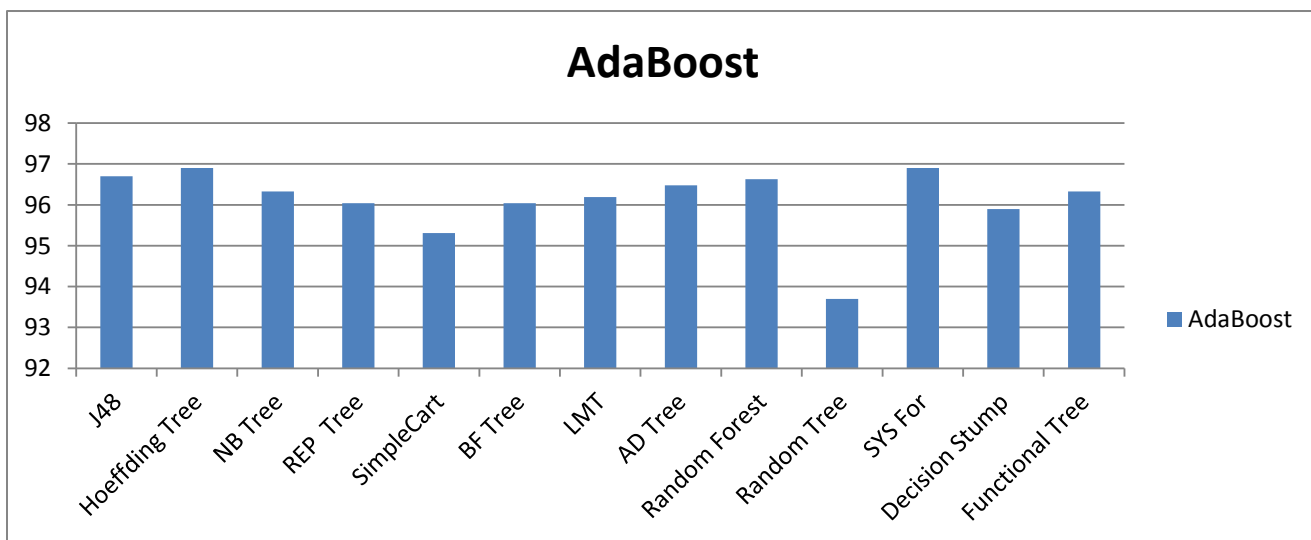


Figure 3 Base classifiers used in AdaBoost vs Accuracy %

3.2.2 Bagging

The bagging classifier was run with 13 different base classifiers in weka. Accuracy of classifications for various base classifiers used in bagging is shown in the table . When the whole data set was used for testing Random Tree was seen to give the highest accuracy of 99.85% followed by Random Forest, with 98.97%. When 10 cross validation was used Hoeffding trees gave the highest accuracy of 97.51% followed by Naive Bayes Tree(NBTree)

with 97.3%. The NBTree model was slower when compared to other classifiers. Time taken by NBTree for building the model was 8.42 seconds. Similarly, LMT classifier is seen to be the next slower classifier followed by functional trees and BF trees. Performance measuring metrics like Kappa Statistic MCC, Precision, Recall, F- measure, ROC area and Confusion Matrix attained for each of the classifiers is given in table 5.

REPTree	95.90	0.01sec	0.9101	0.910	0.959	0.959	0.959	0.981	429 15 13 226
J48	95.90	0.01sec	0.9101	0.910	0.959	0.959	0.959	0.986	429 15 13 226
ADTree	95.46	0.06sec	0.9001	0.900	0.955	0.955	0.955	0.990	429 15 16 223
BFtree	95.02	1.28 sec	0.8908	0.891	0.950	0.950	0.950	0.979	426 18 16 223
Decsion Stump	88.87	0.01 sec	0.7675	0.778	0.904	0.889	0.891	0.950	379 65 11 228
Hoeffding Tree	97.51	0.03 sec	0.9458	0.946	0.976	0.975	0.975	0.993	431 13 4 235
NBTree	97.36	8.42	0.9425	0.943	0.974	0.974	0.974	0.992	431 13 5 234
Random Forest	96.63	0.3 sec	0.9261	0.926	0.966	0.966	0.966	0.992	432 12 11 228
Simple Cart	95.02	1.3 sec	0.8908	0.891	0.950	0.950	0.950	0.978	426 18 16 223
SysFor	96.48	0.37sec	0.9229	0.923	0.965	0.965	0.965	0.993	431 13 11 228
Random Tree	96.33	0sec	0.9195	0.919	0.963	0.963	0.963	0.985	432 12 13 226
LMT	96.48	7.27 sec	0.9228	0.923	0.965	0.965	0.965	0.992	432 12 12 227
Functional Tree	96.33	2.06 sec	0.9196	0.920	0.963	0.963	0.963	0.991	431 13 12 227

Table 5 Performance Metrics of Various Classifiers used in Bagging

3.2.3 Rotation Forest

Performance of 12 base classifiers with Principal Component Analysis method is shown in table 6. Each model is measured using performance metrics Accuracy, Kappa Statistic, MCC, Precision, Recall, F- measure, ROC

area and Confusion Matrix. The best recall and F- measure were given by the base classifiers J48, random Forest and SYSFor and the worst by Hoeffding Trees. Highest Precision was demonstrated by Random Forest Classifier followed by J48 and SYsFor Classifiers.

Table 6 Performance Measures of Classifiers used in Rotation Fores

Base Classifiers used	Accuracy using 10 CV	Time to build the model	Kappa Statitic	Matthews Co relation Coefficient	Precision	Recall	F- measure	ROC Area	Confusion Matrix
REPTree	97.21	0.55 sec	0.9394	0.940	0.973	0.972	0.972	0.990	430 14 5 234
J48	97.36	0.68sec	0.9426	0.943	0.974	0.974	0.974	0.990	430 14 4 235
ADTree	97.21	2.73sec	0.9395	0.943	0.973	0.972	0.972	0.993	429 15 4 235
BFtree	97.21	3.66 sec	0.9395	0.943	0.973	0.972	0.972	0.991	429 15 4 235
Decsion Stump	96.48	0.46 sec	0.9241	0.926	0.967	0.965	0.965	0.988	423 21 3 236
Hoeffding Tree	96.04	1.46 sec	0.9152	0.918	0.964	0.960	0.961	0.991	418 26 1 238
Random Forest	97.36	2.69 sec	0.9428	0.944	0.975	0.974	0.974	0.995	428 16 2 237
Simple Cart	96.92	2.69 sec	0.933	0.933	0.970	0.969	0.969	0.991	429 15 6 233
SysFor	97.36	10.19 sec	0.9426	0.943	0.974	0.974	0.974	0.994	430 14 4 235
Random Tree	97.07	0.28 sec	0.9361	0.936	0.971	0.971	0.971	0.992	430 14 6 233
LMT	96.92	6.74 sec	0.9327	0.933	0.970	0.969	0.969	0.992	431 13 8 231
Functional Trees	96.92	1.45 sec	0.9326	0.933	0.969	0.969	0.969	0.993	432 12 9 230

Among ensemble methods it can be seen that Rotation Forest ensemble provides the best prediction accuracy.

Besides, when compared with single decision trees it can be seen that Rotation Forest also

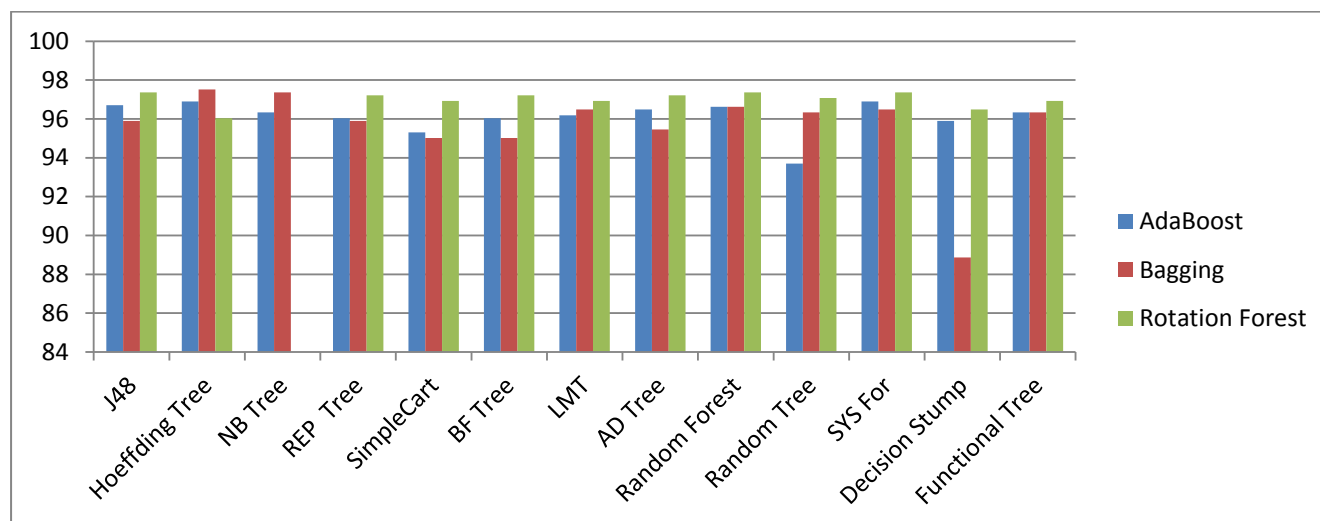


Figure 4 Ensembles with various base classifiers vs Accuracy

outperforms them. Figure 4 shows a comparison of Adaboost, Bagging algorithms and Rotation Forests on basis of their prediction accuracy. In almost all classifiers, except Hoeffding trees, best performance was given by Rotation Forest. In Hoeffding trees, bagging was seen with better performance.

4. CONCLUSION

On evaluation of the performance of various simple decision tree classifiers Naive Bayes Tree classifier was seen providing the best accuracy value among all the single decision tree classifiers. In case of ensemble classifiers Rotation forest provided the best accurate predictions. One drawback with Principal Component Analysis method used

in Rotation Forests is that it computationally expensive. Hence other methods can be explored and used to overcome this problem. Other ensemble methods such as Voting ensembles also help in combining multiple classifiers and produce promising results in prediction. Analysis of the classifiers using other publicly available datasets is to be done to observe if the performance is consistent which will be the future work.

ACKNOWLEDGMENT

The breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

REFERENCES

- [1] Yan-yan SONG, Ying LU, Decision tree methods: applications for classification and prediction, Shanghai Archives of Psychiatry, 2015 Apr 25; 27(2): 130–135.
- [2] Nusaibah Kh. Al-Salihi, Turgay IBRIKCI Classifying Breast Cancer by Using Decision Tree Algorithms, Proceedings of the 6th International Conference on Software and Computer Applications, ACM, Pages 144-148
- [3] Puneet Yadav, Rajat Varshney, Vishan Kumar Gupta, Diagnosis of Breast Cancer using Decision Tree Models and SVM, International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, Volume: 05 Issue: 03 | Mar-2018
- [4] Dr. Neeraj Bhargava, Girja Sharma Dr. Ritu Bhargava, Manish Mathuria., Decision Tree Analysis on J48 Algorithm for Data Mining, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013
- [5] Gaganjot Kaur, Amit Chhabra, Improved J48 Classification Algorithm for the Prediction of Diabetes, International Journal of Computer Applications (0975 – 8887) Volume 98 – No.22, July 2014
- [6] Sweta Rai, Priyanka Saini, Ajit Kumar Jai, Model for Prediction of Dropout Student Using ID3 Decision Tree Algorithm, International Journal of Advanced Research in Computer Science & Technology (IJARCSST 2014), Vol. 2 Issue 1 Ver 2 Jan-March 2014
- [7] Ahmed Iqbal Pritom, Md. Ahadur Rahman Munshi, Shahed Anzar Sabab, Shihabuzzaman Shihab, Predicting Breast Cancer Recurrence using effective Classification and Feature Selection technique, 19th International Conference on Computer and Information Technology, December 18-20, 2016
- [8] Dr. Nalini C, D.Meera, Breast cancer prediction system using Data mining methods, International Journal of Pure and Applied Mathematics Volume 119 No. 12 2018, 10901-10911
- [9] Rawaa Abdulridha Kadhim, Classification and Prediction of Breast Cancer Risk Factors Using Id3, The International Journal Of Engineering And Science (IJES), Volume 5 Issue 11, 2016, PP 29-33
- [10] Sung-Hyuk Cha, Charles Tappert, A Genetic Algorithm for Constructing Compact Binary Decision Trees, JOURNAL OF PATTERN RECOGNITION RESEARCH 1 (2009) 1-13

- [11] Emina Alickovic , Abdulhamit Subasi, Breast cancer diagnosis using GA feature selection and Rotation Forest, Neural Computing and Applications 2015
- [12] Li Min Wang, Xiao Lin Li, Chun Hong Cao, Sen Yuan Miao, Combining decision trees and naive bayes for classification, Elsevier, Knowledge Based Systems 19 (2006) , 511-515
- [13] Yoav Freund, Lew mason, The alternating decision tree algorithm, Proceeding of the Sixteenth International Conference on Machine Learning, Bled, Slovenia, (1999) 124-133.
- [14] E. Venkatesan, T. Velmurugan, Performance Analysis of Decision Tree Algorithms for Breast Cancer Classification, Indian Journal of Science and Technology, Vol 8(29), , November 2015
- [15] Mohamad Badr Al Snousy, Hesham Mohamed El-Deeb, Khaled Badran , Ibrahim Ali al Khlil, Suite of decision-tree based classification algorithms on cancer gene expression data, Egyptian Informatics Journal (2011) pages 73-82
- [16] Kapil Juneja, Chhavi Rana, An improved weighted decision tree approach for breast cancer prediction, International Journal of Information Technology, Springer
- [17] Ajay Kumar Mishra, Bikram Kesari Ratha, Study of Random Tree and Random Forest Data Mining Algorithms for Microarray Data Analysis , International Journal on Advanced Electrical and Computer Engineering, Volume -3, Issue -4, 2016
- [18] Juan J. Rodriguez, Ludmila I. Kuncheva, Carlos J. Alonso (2006). Rotation Forest: A new classifier ensemble method. IEEE Transactions on Pattern Analysis and Machine Intelligence. 28(10):1619-1630
- [19] G. Rasitha Banu , A Role of decision Tree classification data Mining Technique in Diagnosing Thyroid disease, International Journal of Computer Sciences and Engineering, Volume-4, Issue-11, 2016. pp 111-115
- [20] Arundathi A, K Glory Vijayselvi, V Savithri, Assessment of Decision tree Algorithms on Student's Recital, International Research Journal of Engineering and Technology, Volume: 04 Issue: 03 | Mar -2017, pp 2342- 2348
- [21] J.K.F. Bindhia, Yellepeddi Vijayalakshmi, P. Manimegalai, Suvanam Sasidhar Babu, Classification Using Decision Tree Approach towards Information Retrieval Keywords Techniques and a Data Mining Implementation Using WEKA Data Set , International Journal of Pure and Applied Mathematics Volume 116 No. 22 2017, 19-29
- [22] Priyanka Sharma, Comparative Analysis of various Decision tree algorithms using Weka, International Journal on Recent and Innovation Trends in Computing and Communication, Volume 3 Issue 2, 2014, pp 684-690
- [23] Ronak Sumbaly, N. Vishnusri, S. Jeyalatha Diagnosis of Breast Cancer using Decision Tree Data Mining Technique , , International Journal of Computer Applications (0975 – 8887) Volume 98– No.10, July 2014, pp 16-24
- [24] J. Gama, Functional Trees for Classification, Proceedings 2001 IEEE International Conference on Data Mining pp 147-154



A Logistic Regression with Recursive Feature Elimination Model for Breast Cancer Diagnosis

Tina Elizabeth Mathew

Research Scholar, Faculty of Applied Science and Technology,
University of Kerala, Thiruvananthapuram, Kerala, India.

(Corresponding author: Tina Elizabeth Mathew)

(Received 05 June 2019, Revised 20 August 2019 Accepted 30 August 2019)

(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: Cancer also called malignant tumor or malignant neoplasm is one of the world's deadliest diseases. It is the abnormal growth of cells which has the capability of spreading to other parts of the body. Not all tumors are malignant some are benign and they do not invade surrounding cells. Cancer of the Breast is a deadly disease in women. Logistic Regression a binary classifier is used to predict breast cancer. Feature selection methods are employed to find whether reduction of the number of features of the dataset are effective in prediction of Breast cancer. Recursive feature elimination helps in ranking feature importance and selection. The optimal feature sets are selected for building the model using recursive feature elimination with and without cross validation. Recursive feature elimination is used to show the accuracy in prediction when different combinations of features are used based on their ranks. The study shows that reduction of features using RFE helps in improving prediction accuracy.

Keywords: Data Mining(DM), Recursive Feature elimination (RFE), Logistic Regression(LR), Recursive Feature elimination with cross validation (RFECV), Wisconsin Breast Cancer Database(WBCD).

I. INTRODUCTION

Breast Cancer is the second most leading malignancy in the world. It is now the most common cancer in cities and stands second in rural areas in India. Early detection plays a key role in the diagnosis, treatment, prognosis and survivability of the disease. Data Mining(DM) defined as extraction of information from large data sets and part of knowledge discovery in databases(KDD) involves exploring and analyzing large quantities of data and identifying new, valid and useful information from repositories[16]. It has wide applications in market analysis, anomaly detection, medical diagnosis, business analysis and many more. Data mining adopts its techniques from statistics, machine learning, database systems, rough sets, visualization and neural networks. Data mining and statistical techniques can be applied to find useful patterns to help in the important tasks of medical diagnosis and treatment [8].

Data mining strategies are categorized into supervised and unsupervised learning. In supervised learning models, values of inputs are used to make predictions about a target variable with known values. Unsupervised learning models help in predicting on data for which the target variable has unknown values. Data mining models are classified into predictive and descriptive models [16]. Predictive models help in making inferences and forecasting. Descriptive models help to reveal patterns by grouping events, identifying relationships and finding links between events. The tasks included in the Predictive data mining models are prediction and classification Prediction algorithms help to predict continuous or discrete target values from

given input data. Prediction models decide the future outcome rather than existing behavior. Some Predictive models using supervised learning are Regression, Neural Networks, Decision trees, memory based reasoning, and Support Vector Machine. Comparison of these models [25] in heart disease prediction show the efficacy of these models. Similar studies in breast cancer diagnosis using Support Vector Machines [26] also show promising results in prediction when feature elimination was done.

Logistic Regression has been used in many studies in diagnosing breast cancer. In this study, Feature selection and elimination of irrelevant features is applied with Logistic Regression and the significance of using Logistic Regression models alone and in combination with recursive feature elimination techniques with and without Cross validation for breast cancer prediction is analyzed. In section II, the first segment illustrates the related work in this field followed with the materials and methodology used and Section III explains the experimental setup along with the results, Section IV and V gives the Conclusion and Future Scope respectively.

II. MATERIALS AND METHODS

A. Related Work

Ahmed *et al* [1] used Logistic Regression to predict breast cancer. The model selected variables with least correlation and used it to build the LR model. Pearson and deviance statistics were used to measure how closely the model fits the observed data. The model gave an accuracy of 98.9%. Wang *et al* [2] used logistic regression to identify significant factors in hypertension, A neural network with back propagation algorithm was

developed using these significant factors to predict hypertension. This model was seen effective in the prediction of the disease. Haq *et al* [3] used three feature selection algorithms, Relief feature selection, minimal redundancy maximal relevance feature selection algorithm and Least absolute shrinkage and selection operator on seven classifiers, Logistic Regression, KNN and SVM and studied the impact of each methods. It was concluded that feature election helped in classification accuracy and reduced execution time. Choudhury *et al* [4] used Logistic Regression for diagnosis of early stage symptoms of mesothelioma disease and found it to be effective in prediction giving a prediction accuracy of 81.4% in the training set and 63.46% in the testing set. Leopord *et al* [5] in their work used data mining techniques to predict disease outbreak. They suggested that hybrid methods provided better prediction compared with individual classification and regression methods. Bhatti *et al* [6] in their study found that Logistic Regression was an effective method in predicting risk of ischemic heart disease. It was used to assess the risk factors that enhanced the disease risk. Sultana *et al* [7] in their work compared the efficiency of different classifiers, Simple Logistic regression, MLP, Multi-Class Classifiers, DT, REP tree, K-star, IBK, Decision table, PART and Random Forest. Results concluded that Simple Logistic regression method gives the best model in predicting breast cancer. Results indicated that Simple Logistic regression obtained best performance in general compared to the other classifiers in terms of classification accuracy, RMSE, specificity and sensitivity, F-measure, ROC curve area, time taken to build the model and Kappa Statistics. Chang *et al* [9] used Bayesian LR to predict breast cancer. Since the dataset has multicollinear variables, they were avoided based on scores of the variance inflation factor(VIF). Variables with high VIF values were avoided. The model produced an F1 score of 0.95. Mythili *et al* [10] proposed that combinations of support vector machines, logistic regression, and decision trees helped in an accurate prediction of heart disease. Hasan *et al* [11] evaluated the prediction performance of neural networks using different techniques. These when compared with the performance of Logistic Regression it was seen that neural networks performed better. Rahimloo *et al* [12] in their work used Artificial Neural Networks and Logistic Regression models to evaluate performance in predicting diabetes. A hybrid model was later constructed using ANN and Logistic regression and it was seen that the error in prediction was less in the hybrid model than that of individual models. Johnson *et al* [13] used Genetic algorithms to find the best feature set which gave better accuracy in prediction of Alzheimer's disease using logistic regression. Yadav *et al* [14] compared three methods Logistic Regression, Multi-layer Perception and Sequential Minimal Optimization algorithms for predicting heart disease. Logistic regression was seen to give the best F measure. Rajbharath *et al* [15] in their work proposed a hybrid of Random Forest and Logistic Regression algorithms for building a breast cancer survivability prediction model.

The Random Forest Technique is used to perform a preliminary screening of variables and to rank them. Then, the new data set based on the top-k important predictors and is used as input into the Logistic Regression model for predicting breast cancer survivability. Rani *et al* [17] in their paper used Logistic regression to preprocess data and eliminate outliers. This helped in increasing the prediction accuracy of heart disease. Sharanyaa *et al* [18] in their paper concluded that Logistic Regression gave an accuracy of 82% in predicting Parkinson's disease over 71% of Random Forest and 94% of K-Nearest Neighbor models. Javali *et al* [19] in their paper compared multiple logistic regression models to evaluate the effectiveness of the models in identifying risk factors for dental caries and periodontal disease. Using a reduced set of risk factors in the logistic model was found to give better predictions. Shravya *et al* [20] compared various models and found that Logistic Regression, Support Vector Machine (SVM) and K Nearest Neighbor (KNN) for Breast cancer prediction and found that SVM model gave the best prediction accuracy. Gai *et al* [21] in their paper found that Logistic Regression model had good prediction accuracy, satisfactory accuracy and strong robustness in diagnosis of Hepatobiliary Disease. Liang *et al* [22] in their work compared diagnostic performance between back propagation artificial neural networks (BP ANNs) and Logistic regression (LR) models in predicting the prognosis of acute ischemic stroke. Both methods were found to be promising, while ANN's showed better performance comparatively. Yusuf *et al* [23] applied Logistic regression analysis on the variables from the mammogram results and found the variables and combination of variables that had an impact on identifying breast cancer. Leena *et al* [24] in their survey show cased the necessity of combining two or more data mining methods for better performance in disease prediction.

B. Dataset

The Wisconsin Breast Cancer Database (WBCD) obtained from Dr. William H. Wolberg of Wisconsin University Hospitals, Madison is used. The Original data set contains 699 instances with 11 attributes each. The first attribute the ID of an instance, is discarded as it has no role in prediction, and the next 9 represent different characteristics of an instance. These are the cytological characteristics of the breast fine needle aspiration(FNA) test. The instances all have a value between 1 and 10. 1 for benign and 10 for the most malignant. The diagnosis made is the last attribute. Each instance belongs to one of the 2 possible classes, benign with value 2 or malignant with value 4. The 9 attributes that are used in the prediction process are: Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, and Mitoses. The database has 699 instances with 458 benign cases - 65.5% and 241 malignant cases- 34.5%. Sixteen instances were avoided due to missing values and 683 instances were taken for the study, of which 444 belong to benign class and 239 to malignant class.

C. Methodology Used

Regression. Regression is a machine learning technique that determines the relationship between the dependent variable, the target variable whose value is to be predicted, and one or more independent variables. There are three types of regression models: linear, polynomial, and logistic regression. Linear and Polynomial regression uses numeric continuous variables. Logistic regression makes use of categorical dependent variables [11].

Logistic Regression(LR). Logistic regression introduced by David Cox in 1958, is used in predicting binary problems. Consider a dataset containing N points. Each point i consists of a set of m input variables $x_{1,i} \dots x_{m,i}$ which are called independent variables (or predictor variables, features, or attributes), and a binary outcome variable Y_i (or dependent variable, response variable, output variable, or target class). It can assume only the two possible values 0 for failure or 1 for success. A sigmoid function is used in logistic regression to squash the values within the range of $[0,1]$. When the value is greater than a threshold value it is assigned label 1, otherwise it is assigned label 0. The goal of logistic regression is to use the dataset to create a predictive model of the outcome variable. The logistic function is defined in equation 1

$$\sigma(t) = \frac{1}{1+e^{-t}} \quad \text{or} \quad \ln\left(\frac{p}{(1-p)}\right) = t \quad (1)$$

Logistic regression gives probability value, $Y=1$ if malignancy is diagnosed and gives value $Y=0$ for a benign condition. The conditional probability or likelihood that a person has the disease can be computed as $P(Y=1|X)$ Where X represents the set of attributes, $\{x_0, x_1, x_2, x_3, \dots, x_n\}$ that are used in diagnosis and the equation can be given as a linear combination of the inputs as

$$\log\left(\frac{P(X)}{1-P(X)}\right) = x_0 + a_1x_1 + a_2x_2 + \dots + a_nx_n \quad (2)$$

where a_1, a_2, \dots, a_n are the coefficients of the attributes x_1, x_2, x_n and act as weights that imply significance.

Feature Importance and Selection. A dataset has lots of features however, not all features, contribute to the

prediction variable. Removing features of low importance improves accuracy, reduces both model complexity and over fitting and also training time of large datasets. The 9 features of the WBCD data set are Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, and Mitoses. Recursive feature elimination (RFE) is a feature selection method that fits a model and removes the weakest feature or features and keeps repeating the process with the remaining features until the specified number of features is attained or exhausted. It removes features, builds a model using the remaining attributes and calculates model accuracy. Features are ranked relatively according to the order of elimination. A significance level is chosen and the model is fit with all attributes. The attributes with highest p -value is selected and if the p -value is greater than the significance level it is discarded. The model is again built over the remaining attributes. This is repeated till the removal of an attribute affects the accuracy of the built model. Ranking is done based on the coefficient values of the attributes. Higher the coefficient value better is its rank. RFE is able to work out the combination of attributes that contribute mainly to the prediction of the target variable or class.

In this study the value of each attribute is computed using RFE and the features are ranked according to importance. These were explored to find the most prominent and dominating features. The higher the score the more important are the attributes. Studies show that a training partition between 40% to 80 % gives good results in precision and accuracy. The attribute ranking in these training data portions is done to observe whether ratio of training data used has an effect on the relevance of attributes and selection. The values of the features obtained from the classifier is shown in Table 1. The study among partitions suggests slight rank variations but that most prominent features seen in almost all cases are uniformity in cell shape and bare nucleoli, and least being epithelial size.

Table 1: Attribute ranking.

SI . No	Attributes	Feature Ranks based on RFE in				
		70-30 partition	60-40 partition	50-50 partition	80-20 partition	No partition
1	clump_thickness	5	5	6	3	5
2	size_uniformity	1	8	9	1	8
3	shape_uniformity	2	1	1	5	1
4	marginal_adhesion	4	2	2	7	7
5	epithelial_size	9	9	5	9	9
6	bare_nucleoli	3	3	3	2	2
7	bland_chromatin	6	6	7	4	4
8	normal_nucleoli	7	4	4	8	6
9	mitoses	8	7	8	6	3

III. RESULTS AND DISCUSSION

The data is partitioned into training and testing sets feature selection is done and the logistic regression model is used. Feature selection is done in two ways using the RFE and RFECV methods. In RFE, the preferred number of subset of attributes required are

selected at first and it recursively selects subsets of features based on importance. The estimator is first trained on the initial set of features and the importance of each feature is obtained. The least important features are pruned from current set of features and the procedure is recursively repeated on the pruned set until

the specified number of features to be selected is eventually reached.

RFECV does feature ranking with recursive feature elimination and 10 fold cross-validation and selects the optimal number of features and builds the model based on this optimal subset of features. 10 fold Cross-validation divides the samples into a training set and a testing set. The algorithm learns from the training set to constitute the classification rules, the samples of the testing data are used to measure the performance of the classification rules created. All the samples are randomly divided into 10-folds. A fold of the data is used as the testing data and the remaining 9 folds are used as the training set. The step is repeated 10 times, and each testing set validates the classification rules learnt from the corresponding training set to achieve an

accuracy rate. The average of the accuracy rates of all 10 testing sets can be used in the final evaluation results. The subset of features are used in building the model. The model is trained on the training sets and then tested on the testing set. Different number of features are used to build the LR-RFE models and the prediction accuracy of each is assessed. The dataset is partitioned into four groups of training - testing sets in the ratios 70%-30%, 60%-40% 50%-50%, 80%-20% and the performance is accessed. The features are in the order 'clump_thickness', 'size_uniformity', 'shape_uniformity', 'marginal_adhesion', 'epithelial_size', 'bare_nucleoli', 'bland_chromatin', 'normal_nucleoli', 'mitoses'. The work was done using Python Programming using Scikit learn packages. The block diagram in figure 1 represents the working of the model.

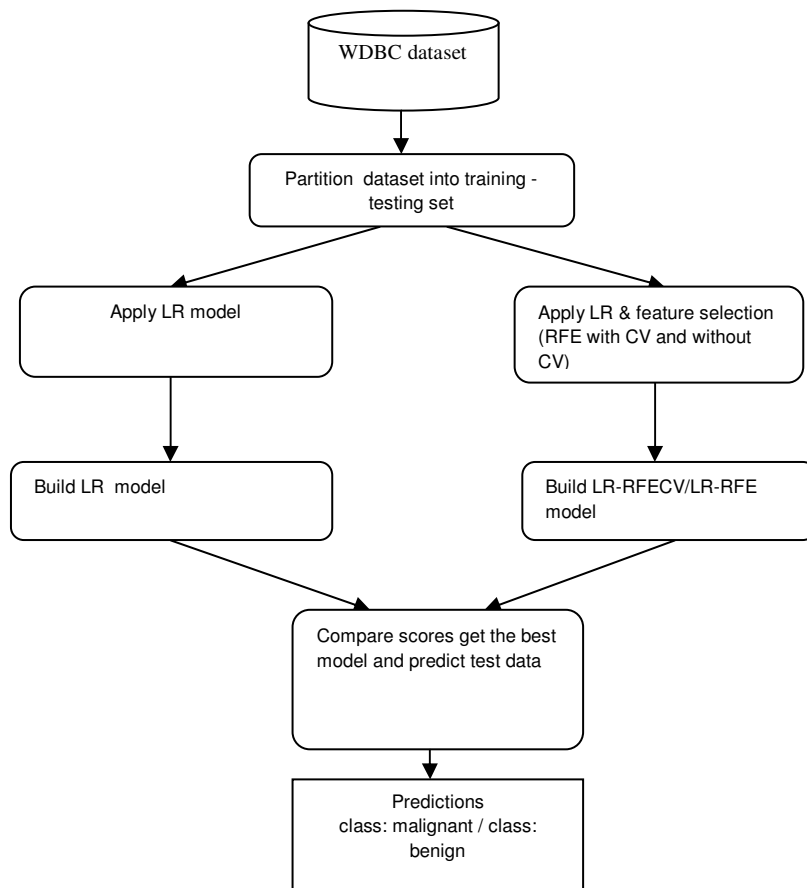


Fig. 1. Block diagram of the model.

A. RFE with cross validation (RFECV)

RFE with 10 fold cross validation(RFECV) was used and the optimal number of features are selected. Graphs for all training- testing partitions were produced and as seen in the concerned figures the optimal number of features selected against cross validation is shown. The model is built with these selected optimal features.

For the training-testing partition of 70-30 the optimal features were 4 and the graph of figure 2 shows the cross validation scores plotted against number of features selected. The graph has its peak values when 4 attributes are selected.

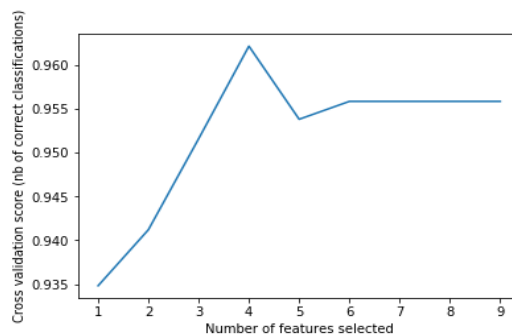


Fig. 2. CV vs Number of Features selected.

For the training-testing partition of 50-50 the optimal features were 3 and the cross validation scores plotted against number of feature selected was attained as shown in Fig. 3. The model is built using these three features.

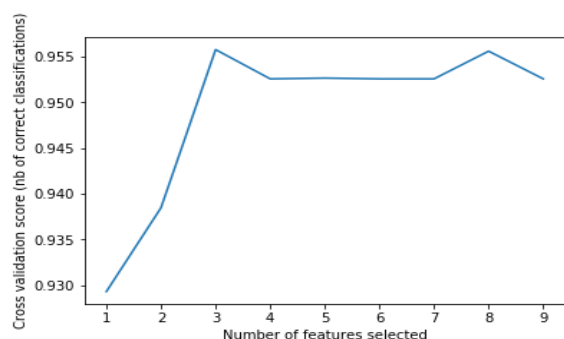


Fig. 3. CV score vs Number of Features selected.

For the training-testing partition of 60-40 the optimal features were 8 and the graph in Fig. 4 shows the cross validation scores plotted against number of feature selected. The model is then built using these 8 features. For the training-testing partition of 80-20 the optimal features were 7 and the graph in Fig. 5 shows the cross validation scores plotted against number of feature selected. Model with the selected 7 features are built. It lies within the range[-1,1]. A -1 value indicates wrong classifications by classifier.

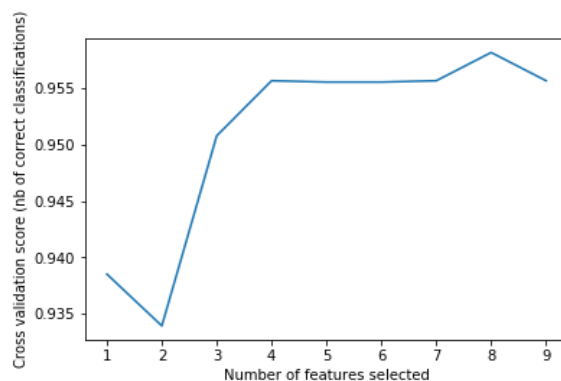


Fig. 4. CV score vs Number of Features selected.

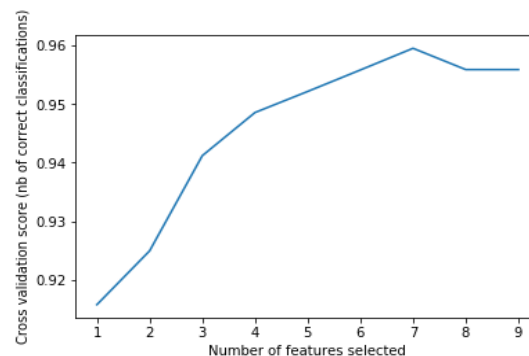


Fig. 5. CV score vs Number of Features selected.

Table 2: Performance scores in 70-30 partition.

SI No	No. of Features used in the model	Features used	Precision = TP/(TP+FP)	Sensitivity / Recall	F1 score	Confusion matrix	MCC	No of Instances in each set(70-30) class 2 - benign and 4- malignant
1	1	[5 1 2 4 9 3 6 7 8]	0.93	0.93	0.92	[[130 0] [15 60]]	0.847	2-130
2	2	[4 1 1 3 8 2 5 6 7]	0.92	0.92	0.91	[[129 1] [16 59]]	0.825	4-75
3	3	[3 1 1 2 7 1 4 5 6]	0.95	0.95	0.95	[[130 0] [10 65]]	0.917	
4	4	[2 1 1 1 6 1 3 4 5]	0.95	0.95	0.95	[[129 1] [9 66]]	0.923	
5	5	[1 1 1 1 5 1 2 3 4]	0.96	0.96	0.96	[[129 1] [7 68]]	0.952	
6	6	[1 1 1 1 4 1 1 2 3]	0.97	0.97	0.97	[[129 1] [6 69]]	0.966	
7	7	[1 1 1 1 3 1 1 1 2]	0.97	0.97	0.97	[[129 1] [5 70]]	0.980	
8	8	[1 1 1 1 2 1 1 1 1]	0.97	0.97	0.97	[[129 1] [5 70]]	0.980	
9	9	[1 1 1 1 1 1 1 1 1]	0.97	0.97	0.97	[[129 1] [5 70]]	0.980	

A +1 value indicates a perfect classification and value near 0 indicates random predictions. Matthews Correlation Coefficient(MCC) is calculated and taken into consideration. Maximum MCC value attained here is 0.98 for 7, 8 and 9 feature sets.

Table 3 gives the performance of the 60-40 partition datasets. In the 60-40 partition the following results were seen. The model with 8 attributes and 9 attributes

gave an F1 score of 0.97. The models using 4, 5, 6, 7 features respectively had an F1 score of 0.96 but the number of false negatives varied. MCC value is maximum at 0.963.

The results of the 50 -50 partition is seen in table 4. The 50-50 set showed the following results. The highest F1 score was attained when 4 features were used. But best MCC value was obtained for model with 5 and 7

features. Table 5 is the result for the 80-20 partitioned dataset. In the 80-20 training-testing partition set the models using 8 and all 9 features gave an f1 score of 0.99 and MCC of 1. Comparing the ratio of the testing datasets for the four partitions it is seen that

performance scores when feature selection is applied on each testing set varies when the percentage of the testing dataset is altered. This is due to the nature of the dataset.

Table 3: Performance scores in 60-40 partition.

SI No	No. of Features used in the model	Features used	Precision =TP/(TP+FP)	Sensitivity/ Recall	F1 score	Confusion matrix	MCC	No of Instances in each ser(60-40) class 2 - benign and 4-malignant
1	1	[5 8 1 2 9 3 6 4 7]	0.91	0.91	0.90	[[165 5] [21 83]]	0.799	2-170
2	2	[4 7 1 1 8 2 5 3 6]	0.92	0.91	0.91	[[167 3] [21 83]]	0.811	4-104
3	3	[3 6 1 1 7 1 4 2 5]	0.95	0.95	0.95	[[169 1] [12 92]]	0.913	
4	4	[2 5 1 1 6 1 3 1 4]	0.96	0.96	0.96	[[169 1] [10 94]]	0.933	
5	5	[1 4 1 1 5 1 2 1 3]	0.96	0.96	0.96	[[169 1] [11 93]]	0.923	
6	6	[1 3 1 1 4 1 1 1 2]	0.96	0.96	0.96	[[169 1] [9 95]]	0.943	
7	7	[1 1 1 1 3 1 1 2 1]	0.96	0.96	0.96	[[169 1] [9 95]]	0.943	
8	8	[1 1 1 1 2 1 1 1 1]	0.97	0.97	0.97	[[169 1] [8 96]]	0.953	
9	9	[1 1 1 1 1 1 1 1 1]	0.97	0.97	0.97	[[169 1] [7 97]]	0.963	

Table 4: Performance score in 50 -50 partition.

SI No	No. of Features used in the model	Features used	Precision =TP/(TP+FP)	Sensitivity/ Recall	F1 score	Confusion matrix	MCC	No of Instances in each set(50-50) class 2 - benign and 4-malignant
1	1	[6 9 1 2 5 3 7 4 8]	0.92	0.92	0.92	[[206 5] [22 109]]	0.834	2-211
2	2	[5 8 1 1 4 2 6 3 7]	0.93	0.92	0.92	[[209 2] [25 106]]	0.825	4-131
3	3	[4 7 1 1 3 1 5 2 6]	0.96	0.95	0.95	[[210 1] [15 116]]	0.908	
4	4	[3 6 1 1 2 1 4 1 5]	0.96	0.96	0.96	[[210 1] [14 117]]	0.916	
5	5	[2 5 1 1 1 1 3 1 4]	0.95	0.95	0.95	[[210 1] [16 115]]	0.900	
6	6	[1 4 1 1 1 1 2 1 3]	0.95	0.95	0.95	[[210 1] [17 114]]	0.893	
7	7	[1 3 1 1 1 1 1 1 2]	0.95	0.95	0.95	[[210 1] [16 115]]	0.900	
8	8	[1 2 1 1 1 1 1 1 1]	0.95	0.95	0.95	[[210 1] [17 114]]	0.893	
9	9	[1 1 1 1 1 1 1 1 1]	0.95	0.95	0.95	[[210 1] [17 114]]	0.893	

A comparison of LR-RFE and LR- RFECV models are shown in table 6, the models show a F1 score varying between the range of 0.95 to 0.98. Precision, Recall and F1 score and MCC of each model is shown in table 6. The recursive feature elimination method ranked features according to importance and it was seen that in all cases the features having the most impact on predictions were the attributes uniformity of cell shape. marginal adhesion and bare & normal nucleoli. MCC scores for LR-RFE was better than LR-RFECV and the

F1 score of LR-RFECV was better than LR-RFE. RFECV method uses the number of optimal features identified by it during cross validation of the data, whereas, the RFE method takes the optimal features as half of the number of features available in the dataset unless specified otherwise. Results in the previous tables 2-7, confirm that better accuracy is obtained by using varied number of features than exactly using half of the available features.

Table 5: Performance score in 80 -20 partition.

SI No	No. of Features used in the model	Features used	Precision =TP/(TP+FP)	Sensitivity/ Recall	F1 score	Confusion matrix	MCC	No of Instances in each set(80-20) class 2 - benign and 4- malignant
1	1	[3 1 5 7 9 2 4 8 6]	0.95	0.94	0.94	[[95 0] [8 34]]	0.864	2-95
2	2	[2 1 4 6 8 1 3 7 5]	0.97	0.97	0.97	[[94 1] [3 39]]	0.980	4-42
3	3	[1 1 3 5 7 1 2 6 4]	0.98	0.98	0.98	[[95 0] [3 39]]	0.991	
4	4	[1 1 2 4 6 1 1 5 3]	0.97	0.97	0.97	[[95 0] [4 38]]	0.966	
5	5	[1 1 1 3 5 1 1 4 2]	0.98	0.98	0.98	[[95 0] [3 39]]	0.991	
6	6	[1 1 1 2 4 1 1 3 1]	0.97	0.97	0.97	[[95 0] [4 38]]	0.966	
7	7	[1 1 1 1 3 1 1 2 1]	0.98	0.98	0.98	[[95 0] [3 39]]	0.991	
8	8	[1 1 1 1 2 1 1 1 1]	0.99	0.99	0.99	[[95 0] [2 40]]	1.0	
9	9	[1 1 1 1 1 1 1 1 1]	0.99	0.99	0.99	[[95 0] [2 40]]	1.0	

Table 6: Comparison of LR-RFECV and LR-RFE models.

SI No	Traini ng- Testin g Set Ratio	LR- RFECV					Confus ion matrix	LR- RFE				MCC	Confusion matrix
		Precis ion	Rec all	F1- Scor e	No. of featu res used	MCC		Precis ion	Reca ll	F1- Score	No. of featu res used		
1	70-30	0.95	0.95	0.95	4	0.903	[[129 1] [9 66]]	0.95	0.95	0.95	4	0.923	[[129 1] [9 66]]
2	50-50	0.96	0.95	0.95	3	0.903	[[210 1] [15 116]]	0.96	0.96	0.96	4	0.916	[[210 1] [14 117]]
3	60-40	0.97	0.97	0.97	8	0.934	[[169 1] [8 96]]	0.96	0.96	0.96	4	0.933	[[169 1] [10 94]]
4	80-20	0.98	0.98	0.98	7	0.949	[[95 0] [3 39]]	0.97	0.97	0.97	4	0.966	[[95 0] [4 38]]

Table 7: Comparison of LR and LR-RFE models.

SI No	Traini ng- Testi ng partiti on ratio used	LR without feature elimination(9 attributes used)					LR-RFE						
		Precis ion	Rec all	F1 score	Confusio n Matrix	MCC	Traini ng- Testi ng partiti on ratio used	No. of attribu tes used in LR- RFE model	Precis ion	Rec all	F1- Scor es	Confusio n Matrix	MCC
1	70-30	0.97	0.97	0.97	[[129 1] [5 70]]	0.937	70-30	7	0.97	0.97	0.97	[[129 1] [5 70]]	0.980
2	50-50	0.95	0.95	0.95	[[210 1] [17 114]]	0.893	50-50	4	0.96	0.96	0.96	[[210 1] [14 117]]	0.916
3	60-40	0.97	0.97	0.97	[[169 1] [7 97]]	0.963	60-40	8	0.97	0.97	0.97	[[169 1] [8 96]]	0.953
4	80-20	0.99	0.99	0.99	[[95 0] [2 40]]	1.0	80-20	8	0.99	0.99	0.99	[[95 0] [2 40]]	1.0

LR (without feature elimination) using all 9 features and LR-RFE models are also compared in table 7. It was seen that LR -RFE obtained the same Precision, Recall and F1 scores for a reduced set of attributes when compared with the scores of LR without feature elimination. In the 70-30 partition, an F1 score of .97 was obtained with LR-RFE using 7 attributes instead of all 9. In 50-50 ratio LR-RFE got a better F1 score of .96 with 4 features. Similarly, 60-40 and 80-20 partitions also showed an F1 score for 0.97 and 0.99 respectively for reduced feature set of 8 attributes. MCC values were better for LR with feature elimination than the individual LR model in all partitions as shown in table 7. Thus supporting the fact that feature elimination and reduction of attributes enhances prediction accuracy. The receiver operating characteristics curve (ROC) is a performance measurement curve which plots sensitivity against specificity. Area Under Curve(AUC) has a value range between 0.5 to 1, where 0.5 denotes a bad classifier and 1 denotes a good classifier. The ROC and AUC for the various models are given as follows
The ROC of the LR-RFECV model is shown in figure 6 and the Area under curve calculated has value 1.

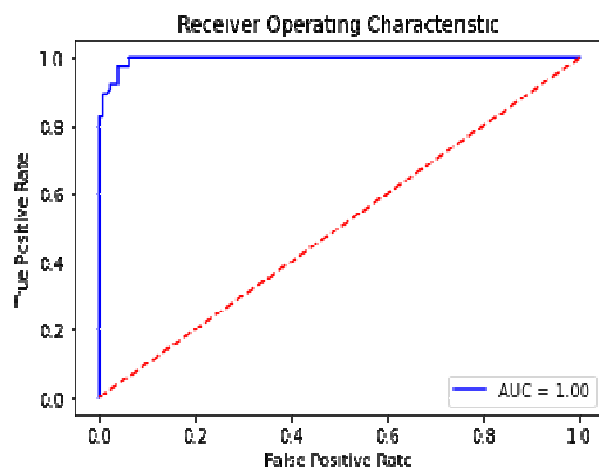


Fig. 6. ROC curve of LR-RFECV.

The ROC for LR-RFE model is shown in figure 7 and it can be seen that the Area Under Curve has value 1.

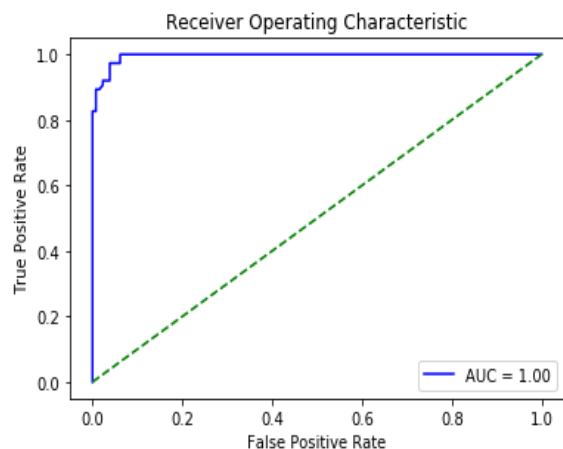


Fig. 7. ROC for LR-RFE.

IV. CONCLUSION

From the work it can be concluded that LR models with feature elimination methods provide better performance than when using the same model without feature elimination. Reduced feature set helps in improving model accuracy. Logistic regression deals effectively with outliers. The study highlights the importance of feature elimination for performance enhancement, in terms of accuracy, in supervised data mining models, hence aiding medical practitioners in easy and quick diagnosis of the disease.

V. FUTURE SCOPE

The future work will be to apply other feature reducing methods on different classifiers and to combine various data mining methods to evaluate their impact and performance enhancement on prediction accuracy in breast cancer diagnosis.

Conflict of Interest. Nil

ACKNOWLEDGMENT

The author is thankful to Dr AnilKumar K S, Associate Professor and Research guide in Technology Management, University of Kerala for his encouragement and support for the work. The breast cancer database was obtained from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

REFERENCES

- [1]. Ahmed F. S., & Shawky, D.M., (2015). Logistic Regression Model for Breast Cancer Automatic Diagnosis, *SAI Intelligent Systems Conference 2015* November 10-11.
- [2]. Wang, A., An, N., Xia, Y., Li, L., & Chen, G., (2014). A Logistic Regression and Artificial Neural Network-based Approach for Chronic Disease Prediction: a Case Study of Hypertension, 2014 IEEE International Conference on Internet of Things (iThings 2014), Green Computing and Communications (GreenCom 2014), and Cyber-Physical-Social Computing (CPSCom 2014).
- [3]. Haq, A. U., Li, J. P., Memon, M. H., Nazir, S., & Sun, R. (2018). A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms. *Mobile Information Systems, 2018*, 1-21.
- [4]. Choudhury, A. (2018). Identification of Cancer-Mesothelioma Disease Using Logistic Regression and Association Rule. *arXiv preprint arXiv:1812.10384*.
- [5]. Leopord, H., Cheruiyot, W. K., & Kimani, S. (2016). A survey and analysis on classification and regression data mining techniques for diseases outbreak prediction in datasets. *Int. J. Eng. Sci, 5*(9), 1-11.
- [6]. Bhatti, I. P., Lohano, H. D., Pirzado, Z. A., & Jafri, I. A. (2006). A logistic regression analysis of the ischemic heart disease risk. *Journal of Applied Sciences, 6*(4), 785-88.
- [7]. Sultana, J. & Jilani, A.K., (2018). Predicting Breast Cancer Using Logistic Regression and Multi-Class Classifiers, *International Journal of Engineering & Technology, 7*(4.20): 22-26.
- [8]. Mangasarian, O. L., & Wolberg, W. H. (1990). *Cancer diagnosis via linear programming*.

University of Wisconsin-Madison Department of Computer Sciences.

- [9]. Chang, M., Dalpatadu, R.J., Phanord, D., & Singh, K.A., (2018). Breast Cancer Prediction Using Bayesian Logistic Regression, *Open Access Biostatistics & Bioinformatics*, Vol. 2, Issue 3, 1-5.
- [10]. Mythili, T., Mukherji, D., Padalia, N., & Naidu, A. (2013). A heart disease prediction model using SVM-Decision Trees-Logistic Regression (SDL). *International Journal of Computer Applications*, 68(16): 11-15.
- [11]. Hassan, M., Butt, M. A., & Baba, M. Z. (2017). Logistic Regression Versus Neural Networks: The Best Accuracy in Prediction of Diabetes Disease. *Asian Journal of Computer Science and Technology*, Vol. 6 No. 2, 2017, pp.33-42.
- [12]. Rahimloo, P., & Jafarian, A. (2016). Prediction of Diabetes by Using Artificial Neural Network, Logistic Regression Statistical Model and Combination of Them. *Bulletin de la Société Royale des Sciences de Liège*, 85, 1148-1164.
- [13]. Johnson, P., Vandewater, L., Wilson, W., Maruff, P., Savage, G., Graham, P., ... & Rowe, C. C. (2014). Genetic algorithm with logistic regression for prediction of progression to Alzheimer's disease. *BMC bioinformatics*, 15(16), S11.
- [14]. Yadav, P. K., Jaiswal, K. L., Patel, S. B., & Shukla, D. P. (2013). Intelligent heart disease prediction model using classification algorithms. *IJCSMC*, 3(08), 102-107.
- [15]. Rajbharath, R., & Sankari, L., (2017). Predicting Breast Cancer using Random Forest and Logistic Regression. *International Journal of Engineering Science and Computing*, Vol. 7, issue 4, pg. 10708-10712.
- [16]. Data Mining Group, <http://dmg.org/pmml/v2-0/Regression.html>
- [17]. Rani, S., K., Manoj, S., M., & S Mani, S.,G., (2018). A heart disease prediction model using Logistic Regression. *International journal of Trend in Scientific Research and Development*, Vol. 2, Issue 3, 1463-1466.
- [18]. Sharanyaa, S., Gunavathiel, M.A., Abitha. P, & Sangeetha, K., (2018). Classification of Parkinson's Disease Using Logistic Regression. *International Journal of Pure and Applied Mathematics*, Volume 118 No. 18, 1587-1593.
- [19]. Pandit, P. V., & Javali, S. B. (2012). Multiple logistic regression model to predict risk factors of oral health diseases. *Romanian Statistical Review*, 5, 1-14.
- [20]. Shravya, Ch., Pravalika, K., & Subhani, S., (2019) Prediction of Breast Cancer Using Supervised Machine Learning Techniques. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 8(6): 1106-1110.
- [21]. Gai, X., & Zhang, Y. (2019). Diagnosis of Hepatobiliary Disease Based on Logistic Regression Model. In *IOP Conference Series: Materials Science and Engineering* (Vol. 490, No. 6, p. 062084). IOP Publishing.
- [22]. Liang, Y., Li, Q., Chen, P., Xu, L., & Li, J. (2019). Comparative study of back propagation artificial neural networks and logistic regression model in predicting poor prognosis after acute ischemic stroke. *Open Medicine*, 14(1), 324-330.
- [23]. Yusuff, H., Mohamad, N., Ngah, U., & Yahaya, A. (2012). Breast cancer analysis using logistic regression. *International Journal of Research and Reviews in Applied Sciences*, 10(1), 14-22.
- [24]. Sarvaiya, L., Yadav, H., & Agrawal, C., (2019). A Literature review of Diagnosis of Heart Disease using Data Mining Techniques, *International Journal of Electrical, Electronics and Computer Engineering*, 8(1): 40-45.
- [25]. Basha, S.M., Bagyalakshmi, K., Ramesh, C., Rahim, R., Manikandan, R. & Kumar, A. (2019). Comparative Study on Performance of Document Classification Using Supervised Machine Learning Algorithms: KNIME. *International Journal on Emerging Technologies*, 10(1): 148-153.
- [26]. Mathew, T.E. (2019). A Comparative Study on the Performance of different Support Vector Machine Kernels in Breast Cancer Diagnosis. *International Journal of Information and Computing Science*, Volume 6, Issue 6, 432-441.

How to cite this article: Mathew, T.E. (2019). A Logistic Regression with Recursive Feature Elimination Model for Breast Cancer Diagnosis. *International Journal on Emerging Technologies*, 10(3): 55–63.

FAULT PREDICTIONS OF A CONSTRAINED COMPLEX SYSTEM USING 1D AND 2D-CELLULAR AUTOMATA – A COMPARISON

Priya R

*Assistant Professor, Department of Computer Science
Government College, Kariavattom,
Trivandrum, Kerala 695581, India
priyanil2007@gmail.com*

Abstract-Functionality based failure analysis and validation during design process in a constrained complex system is challenging. In this paper, we advocate a model to validate the functionality of a constrained complex control system with its structural behavior. An object constrained model is proposed for validation of any component of a complex system with constraints and its state of safeness is predicted using cellular automata. The model consists of two subsystems; an inference engine that functions based on rule based expert system and a failure forecast engine based on cellular automata. The system is tested against the thermal power plant for early detection of failure in the system which enhances the process efficiency of power generation. A comparison is also made between failure predictions based on 1 Dimensional Cellular Automata (1DCA) and 2 Dimensional Cellular Automata (2DCA) and established that 2DCA yields better results.

Keywords-Complex System, Constrained Objects, Cellular Automata, Control System, Prediction Engine, Failure Forecast Engine

I. INTRODUCTION

Any complex system can be mapped to an object-oriented model using class diagram and the constraints using Object Constraint Language (OCL). OCL can be used for representing class invariants, which are conditions of attributes of classes. OCL can also be used for representing constraints on methods of classes as pre and post conditions [OCL Specification, 2003]. But, the main drawback is that only priori defined constraints can be represented using OCL. The model for fault detection would be more optimal if it could predict new faults that are likely to happen.

Hence, a methodology is proposed for fault prediction where new faults are predicted, which are not specified in the OCL system. The proposed model has two engines, an inference engine, which performs fault detection by validating the model of a complex system against priori-defined constraints, and a failure forecast engine, for predicting the faults from the model. Inference engine works on the principle of model validation against the OCL-based constrained system by providing instance values to the model. Here, the OCL-based system acts as a rule-based expert system, where the model is validated against the defined constraints. Failure forecast engine focuses on the prediction of new faults not defined in the rule-base. Here, the state of safeness of the complex system is defined by a finite solution space. This is accomplished using cellular automata (CA), which performs a complete search within the solution space.

In this work, the goal is to model a failure prediction system and to employ cellular automaton to predict the state of a coal-mill model, which is a complex constrained system. In addition, the predicted faulty condition will be updated as a new rule or constraint in the OCL system using a feedback mechanism. Therefore, the two benefits achieved from failure prediction system with constrained object modeling technique serves better, scaling up with significant out-performance.

Hence, in this work, a validation technique against priori-defined constraints defined as a rule-base system and a failure forecast model using automata, as a combined approach, is proposed. The proposed model is applied for identifying the situations when constraints are violated and then stable state prediction is illustrated for the thermal power plant system. The thermal power plant system considered is a complex system, with many sub-systems. Constraints are defined on all the sub-systems

such as, coal storage, pulveriser, boiler, furnace, fan, generator, turbine and drum of thermal power plant as a rule-based system, which defines the backbone of the inference engine.

Since the system is large enough with many sub components, we have considered only the pulveriser sub-system for failure forecasting. Failure forecasting experiments were performed on pulveriser by considering one parameter, raw coal flow (RCF) using 1 Dimensional Cellular Automata (1-DCA) and also with two parameters, raw coal flow (RCF) and mill current (MC). The results were validated by comparing with real plant site data and results reveal that early detection of failure enhances the process efficiency of power generation. The quantitative analysis of the results reveals that the model is well suited for fault detection and fault prediction of a constrained complex system.

II. RELATED WORKS

Faults can be detected by a variety of quantitative or qualitative means. This includes many of the multivariable and model-based approaches [15]. Detection techniques also include simple, traditional techniques for single variables, such as alarms and Statistical Process Control (SPC) measures. Different approaches for fault detection using mathematical models have also been developed [16].

Petrinets are used to analyse QoS of fault tolerant systems [1]. They have also proposed a self healing model using Petri-nets and aspect-oriented programming to improve the quality of fault tolerant systems [2]. A prediction algorithm based on data mining especially for multimedia objects in next-generation digital Earth is proposed [12]. A comparison of the performance of different prediction methods is done by Yue *et al* [17]. The prediction methods employed were Naive Bayes, Logistic, J48 and Bagging. The results reveal that the highest accuracy obtained was only 91.8%.

Cellular Automaton (CA) has also been employed for prediction in fields such as forest fire spreading [4], modeling centralised control in dynamic systems [7] and so on. It is used for simulation of traffic flow with advanced control vehicles and safety systems [5]. Simulating urban expansion and scenario prediction using a cellular automata urban growth model, SLEUTH, through a case study of Karaj City, Iran was done by Sakieh *et al*. [13]. Accurate prediction of splice site in Bioinformatics is yet another area where CA is used [14].

Prediction of faults from the model of a complex system is also highly appreciable as fault handling cases could also be addressed when the system is built. Many control systems like the steam temperature control system of the boiler component of thermal power plant have been developed using genetic algorithms [9]. Many works have been carried out in coal-mill modeling. A six segment coal-model that covers the whole milling process from mill start-up to mill-shut-down is studied [10]. Modeling coal-mill by machine learning with on-site data was also carried out [11]. In all these works, mathematical modeling of the milling process is done and is optimised at an observable level. Genetic Algorithms were used to find an optimal value to the parameters $K_1..K_{17}$ used in the mathematical modeling of a coal-mill in the real thermal power plant site [8]. All these methods were employed only to detect failures from the model of thermal power plant system and failure prediction is not addressed, which is highly relevant for a constrained complex system.

Hence, we employ the principle of Cellular automata for predicting faulty cases of a constrained complex system. CA has the greatest advantage of defining the search space by identifying the rule space, which forms the backbone of the complete search in the search space. Moreover, it has the advantage of the emergence of macroscopic behaviour from local behaviour. Although the capacity of CA to explore complex systems has been well established [18], its capacity to represent real patterns and to predict from the patterns is yet to be proven. Here, an attempt is made to detect and predict faults from the model of a thermal power plant system using cellular automata. Microscopic properties of sub-systems of the complex model are used to detect the macroscopic feature of failure prediction of the model using CA. The identified faults are updated to the OCL system through the feedback mechanism of the control system model.

III.CONSTRAINED OBJECT MODEL PREDICTIVE FEEDBACK CONTROLLER(CO-MPC)

The object constrained model proposed in this work consists of two sub-systems;

- An inference engine for validation of a constrained complex system.
- A failure forecast engine for predicting the state of safeness of the system.

The architecture of **Co-MPc** is depicted in Figure (1).

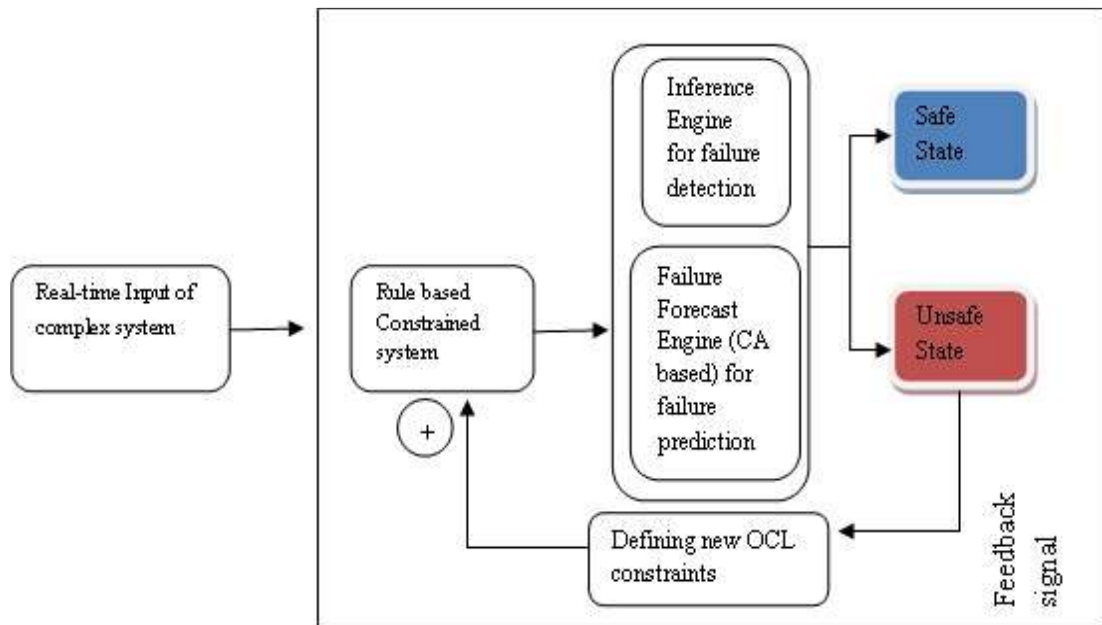


Figure 1 Architecture of Co-MPc

3.1 Inference engine

Inference Engine is a rule-based constrained system that helps to detect fault from the model of a complex system. The model is provided with instance values which are validated based on constraints defined.

The following algorithm detects the secure state of constrained complex system by the inference engine. Here, the OCL system consists of all the invariant constraints defined for all the subsystems of the complex system.

Algorithm 1 Secure state detection by inference engine

Input: Real-time input of the complex system, OCL file with pre-defined constraints

Output: The state assigned to the system (*Safe/Unsafe*)

Safety=true *alarm* = false

repeat at each time t

for each attribute

if val(attribute) \in domain(attribute) in OCL file **then**

Safety=true *alarm*=false

else

Safety=false *alarm*=true

end


```

end

for each dependent attribute pair (attribute1, attribute2)
if val (attribute1) && val (attribute2) complies with OCL file
then

    Safety=true alarm=false
else

    Safety=false      alarm=true
end

end

until alarm==true

```

The rule based inference engine proposed here relies on existing set of constraints for performing the validation. But constraints can be evolutionary, that are automatically incremented to the system. This problem is not addressed effectively with live updation of the fault detected till now, so we proposed a failure forecast model with a feedback mechanism to the OCL based Inference system.

3.2 Failure forecast engine

In large constrained complex systems, constraints can be evolutionary, that can be incorporated effectively using Cellular Automata by simulating the structural characteristics of complex systems. Both the temporal and spatial complexities of systems can be efficiently modeled by precise definition of transition rules in CA models. Failure forecast engine works on the principle of cellular automata. For predicting failures from the models of complex real-time systems, one dimensional or two dimensional automata can be used. One dimensional cellular automaton predicts the state of system from a single parameter, whereas two-dimensional cellular automata use two parameters for prediction. The neighbourhood defined for both 1-DCA and 2-DCA is von Neumann neighbourhood.

The pattern of attribute values of sub-systems can be used as the input data for the prediction of the future pattern of attributes which in turn judges the state of the sub-system. The ground state of the CA is defined based on this input data. This defines the state of each cell at time instant t . The dependency among attributes is the driving force to consider 2-DCA for prediction. The state of each cell changes to the next value at time instant $t + 1$ based on the CA rules. The state of each cell is defined as a set of three values which is a three state automaton: $\{-1, 0, 1\}$

The [value, state] pair is defined as follows:

[0, toggling state] [1, safe state] [-1, unsafe state]

For 1-DCA, just a single parameter or attribute of a sub-system is used for predicting the state of safeness of that sub-system. The rules for defining the ground state from input pattern for 1-DCA is given below.

Rule 1 At instant t , if the domain value of the attribute is within the domain boundary, then the cell state is safe or 1.

Rule 2 At instant t , if the domain value of the attribute is not within the domain boundary, then the cell state is unsafe or -1.

Rules are also defined for the transition of cell states from the ground state, based on von Neumann neighbourhood. Cell states transitions are defined separately for two categories of cells. The categories are:

- (i) cells with one boundary
- (ii) cells with two boundaries

The next state for cells with one boundary and two boundaries are defined based on equation (1).

$$c_x^{t+1} = f(n_x^t)$$

(1)

For cells with one neighbour,

$$n_x^t = \begin{cases} c_{x+1}(t) & x = 1 \\ c_{x-1}(t) & \text{otherwise} \end{cases} \quad (2)$$

$$f: c_x^{t+1} = \begin{cases} 1 & \Sigma n_x^t = 1 \\ 0 & \Sigma n_x^t = 0 \\ -1 & \Sigma n_x^t = -1 \end{cases} \quad (3)$$

For cells with two neighbours,

$$n_x^t = \{c_{(x-1)}(t), c_{(x+1)}(t)\} \quad (4)$$

$$f: c_x^{t+1} = \begin{cases} 1 & \Sigma n_x^t = 2 \\ 0 & \Sigma n_x^t = 1 \\ -1 & \text{otherwise} \end{cases} \quad (5)$$

Any complex system has enormous number of attributes and functionalities, and certain attributes may be dependent on each other. Such dependent parameters can also be used to predict the state of the complex system after 't' instance. Here, two attributes of a subcomponent of a complex system are used for prediction, which defines the 2-DCA.

The ground state $N \times N$ of 2-DCA is defined based on the following three rules.

Rule 1 At instant t, if the domain value of attribute 1 and attribute 2 are within the domain boundary, then the cell state is safe or 1.

Rule 2 At instant t, if either the domain value of attribute 1 or domain value of attribute 2 is not within their domain boundary, then the cell state is toggling or 0.

Rule 3 At instant t, if both the domain value of attribute 1 and attribute 2 are not within their domain boundary, then the cell state is unsafe or -1.

Rules are also defined for the transition of cell states from the ground state, based on von Neumann neighbourhood. Cell states transitions are defined separately for three categories of cells. The categories are:

- (i) cells with four boundaries
- (ii) cells with three boundaries
- (iii) cells with two boundaries

At each time step t, exactly one of three things can happen to a cell with four-boundary cells.

- (i) **Birth** : The cell state becomes 1 (safe) at state t, if $\Sigma C_1 = \{2 \text{ or } 3 \text{ or } 4\}$ at $t - 1$; where ΣC_1 is the summation of the values of the state of the four boundary cells.
- (ii) **Survival**: The cell state becomes 0 (toggling) at state t, if $\Sigma C_1 = \{0 \text{ or } 1\}$ at $t - 1$.
- (iii) **Death**: The cell state becomes -1 (unsafe) at state t, if $\Sigma C_1 = \{-4 \text{ or } -3 \text{ or } -2 \text{ or } -1\}$ at $t - 1$; which implies all the neighbouring cells were either in unsafe state or in toggling state.

CA rules for cells with three boundaries:

- (i) **Birth**: The cell state becomes 1 (safe) at state t, if $\Sigma C_1 = \{2 \text{ or } 3\}$ at $t - 1$; where ΣC_1 is the summation of the values of the state of the three boundary cells.
- (ii) **Survival**: The cell state becomes 0 (toggling) at state t, if $\Sigma C_1 = \{0 \text{ or } 1\}$ at $t - 1$.
- (iii) **Death**: The cell state becomes -1 (unsafe) at state t, if $\Sigma C_1 = \{-3 \text{ or } -2 \text{ or } -1\}$ at $t - 1$; which implies all the neighbouring cells were either in unsafe state or in toggling state.

CA rules for cells with two boundaries:

- (i) *Birth*: The cell state becomes 1 (safe) at state t , if $\Sigma C_1 = 2$ at $t - 1$; where ΣC_1 is the summation of the values of the state of the two boundary cells.
- (ii) *Survival*: The cell state becomes 0 (toggling) at state t , if $\Sigma C_1 = \{-1 \text{ or } 0 \text{ or } 1\}$ at $t - 1$.
- (iii) *Death*: The cell state becomes -1 (unsafe) at state t , if $\Sigma C_1 = \{-2\}$ at $t - 1$; which implies all the neighbouring cells were either in unsafe state or in toggling state.

Cell state transitions based on the defined CA rules are the basis for the generations in cellular automata. From the states obtained after N generations, the state of the sub-system can be predicted, with respect to the two given attributes. Prediction can be made faster by estimating the percentage of unsafe states after each generation. If the percentage of unsafe state cells is above an acceptable limit, then fault is predicted from that generation itself. Thereby, the state of safeness of the whole system is predicted. New rules are identified from the prediction results and are updated to the rule-base of the inference engine as a feedback mechanism.

Algorithm 2 Safe/unsafe state detection by failure forecast engine using 2-DCA

Input: Input data of *attribute1* and *attribute2* captured once in t ms, threshold value of error acceptable (*threshold*), OCL file and number of iterations (N)

Output: The state of the system predicted (*Safe/Unsafe*)

```

Safety=true      alarm = false      no_of_iterations = 1
repeat at each time  $t=t'$  sec
  i=1
  repeat for each i
    j=1
    repeat for each j
      if val(attribute1) ∈ domain(attribute1) && val(attribute2) ∈ domain(attribute2)
        then
          gs(i,j)=1
          j=j+1
        end
      if either val(attribute1) ∈ domain(attribute2) || val(attribute2) ∈ domain(attribute2)
        then
          gs(i,j)=0
          j=j+1
        end
    end
  if val(attribute1) ∉ domain(attribute1) && val(attribute2) ∉ domain(attribute2)
    then
      gs(i,j) = -1
      j=j+1
    end
  until j>5
  i=i+1
until i>5

unsafe_perc = (Σ gs(i,j)/25)*100; where gs(i,j) = -1;
if unsafe_perc > threshold
  then

```

```

Safety=false; alarm=true// Predicted alarm
//OCL file updation through Feedback
k=1          set=0
repeat
if new_val_attribute1=val(attribute1) and new_val_attribute2=val(attribute2)
then
          set=1
        else
          set=0 break
        k=k+1
until k>24
if set=1 then
  alarm=true Safety=false
  call fault handlers
else
  Safety=true alarm=false

end

gsnew=fun_update(gs) // fun_update is the function used to update the values of gs based on the
CA rules defined above

gs=gsnew

no_of_iterations= no_of_iterations+1
until no_of_iterations>N

```

The results obtained from cellular automata are validated on the basis that if the pattern of cell states remains the same for N generations without even a single unsafe state, then the particular subcomponent will work efficiently without failure with respect to the two attributes considered for that sub-system. Moreover, if the pattern changes and if the percentage of unsafe state goes on increasing through generations, then it is predicted that the state of safeness of the system moves to unsafe state.

IV.SIMULATION RESULTS ON TTPS (THERMAL POWER PLANT SYSTEM)

The methodology is implemented in a complex constrained system, TamilNadu Tuticorin Power Station (TTPS), which is a coal-fired thermal power plant system with a number of sub-systems and the system is constrained with respect to the domain of values permitted for the attributes, and the interdependencies among the attributes.

The main sub-systems of a coal-fired thermal power plant are:

- 1 coal supplier
- 2 pulveriser
- 3 boiler
- 4 turbine
- 5 generator
- 6 substation transformer
- 7 condenser
- 8 ash collectors

Each sub-system mentioned has its own attributes and functionalities. The static and dynamic characteristics of the sub-systems are constrained with domain values and pre and post conditions. These components are interdependent on each other with multiplicity constraints. Hence, the whole process represents a constrained complex system.

4.1 Fault Detection on TTPS

Here, the researcher proposes a model to detect faulty conditions from the constrained object model of a coal-fired thermal power plant. Faults are detected by validating the model against the OCL system defined for all the sub-systems of the thermal power plant system. The rule-based system is defined on the following three types of constraints.

1. Domain specific constraints:

The attributes of pulveriser are:

- Raw_Coal_Flow
- Primary_Air_Flow
- Mill_Inlet_Temp
- Primary_Air_Diff_Press
- Mill_Diff_Press
- Mill_Curr

The OCL system, defined for constraints on domain of attributes, is shown below.

- context pulveriser
inv: rawcoalflow<=45
- context pulveriser
inv: coalinlettemp<=300
- context pulveriser
inv: prairdiffpress<=180
- context pulveriser
inv: pairflowrate<=75

2. Inter class constraints: Certain attributes are dependent on each other. They may be of the same class or of different classes. Those dependencies reflect in the domain of dependent attributes. It is also defined in the rule-based system.

- **context** CoalStorage
inv: if a.Stage = 3then
self.Ash=40.4
else
self.Ash=19
end if
- **context** pulveriser
inv:
if CoalHGI=55 and CoalMoisture=0.1 and CoalFineness=0.7
then
if a.Stage=3 then
self.Capacity2=39.9
else
self.Capacity2 =33.8
endif
endif

3. Certain dependent attribute value pairs are necessary for the smooth functioning of the system. It is defined as a triplet as follows:

(< object1>, < attribute1>, < value >)
related to
(< object1>, < attribute2> , < value >)

For instance, (< pulveriser >, < rawcoalflow >, < 25.9...26.60 >)T/hr
and
(< pulveriser >, < millcurrent >, < 33.93...34.90 >)A
are dependent attributes.

These three types of constraints are defined for all the sub-systems of TTPS system. The inference engine defined with these constraints is validated by providing instance values from the plant site. An alarm or a fault is detected if any of the instance values goes below 5% of minimum or above 95% of the maximum value defined as constraint in the OCL system. Table (1) shows the result of validation of inference engine applied to Pulveriser, Boiler and CoalStorage.

Table 1 : Validation results of inference engine

<i>Object</i>	<i>Instance Values</i>	<i>Constraint</i>	<i>Validation Result (safe/alarm)</i>
Pulveriser	Speed=52 Capacity=33.8 CoalFineness=0.7 CoalMoisture=0.1 Rawcoalflow=40 Pairflowrate=70 Coalinlettemp=250 Prairdiffpress=150 outlet_temp=90 diffpr=450 millcurr=50	context Pulveriser inv: Rawcoalflow<=45 Pairflowrate<=75 Coalinlettemp<=300 Prairdiffpress<=180 outlet_temp<=100 diffpr<=500 millcurr<=60	Safe
Boiler	temp=540 Stage=3 RH_Inlet_pr=36.7 RH_Outlet_pr=35.2	context Boiler inv : if a.Stage = 3 then RH_Outlet_pr=35.2 else RH_Outlet_pr=24.5 endif inv : temp=540	Safe
CoalStorage	CalorificValue=5950 FixedCarbon=40.5 Moisture=7.5 Ash=19 Volatility=33 CoalSize=20 HGI=53	context CoalStorage inv cosize: CoalSize=20 hval: HGI=55	Alarm

4.2 Fault Prediction on TTPS

The rule-base of the inference engine is made dynamic by adding new constraints identified by the failure forecast engine. The failure forecast engine, which is implemented using the logic of cellular automata works on the following principle. The state of the system depends on the state of its individual sub-systems. In order to demonstrate that, Pulveriser subsystem of TTPS is considered. Faults are predicted by considering attribute instances of ‘pulveriser’ sub-system of TTPS. The Pulveriser pulverises the coal lumps of 20 mm size to fine granules of less than 8 mm size. The state of safeness of the pulveriser component is predicted based on the values of its attributes at previous time instances. Here, the state of pulveriser is predicted using a single parameter using 1-DCA and also with two parameters using 2-DCA.

4.2.1 Results of Prediction using 1-DCA

One parameter, rawcoalFlow (RCF) of Pulveriser was used for prediction of the state of Pulveriser. In the real thermal power plant, data is captured once in 40 ms. In this work, prediction is based on data captured in 1sec, which gives 25 instance values. These instance values at t_0 to t_{24} time instances form the basis of prediction. Table (2) gives a summary of the domain values of RCF from thermal power plant site.

Table 2: 25 instances of RCF values

Time instance	RCF
t_0	26.11
t_1	26.16
t_2	26.08
t_3	26.16
t_4	25.87
t_5	26.25
t_6	26.22
t_7	26.07
t_8	26.14
t_9	26.14
t_{10}	26.21
t_{11}	26.21
t_{12}	26.21
t_{13}	25.98
t_{14}	25.99
t_{15}	26.07
t_{16}	25.96
t_{17}	26.22
t_{18}	26.12
t_{19}	26.15
t_{20}	26.14
t_{21}	26.16
t_{22}	26.04
t_{23}	26.17
t_{24}	26.16

In the real system of TTPS, the constraint or the most optimal value for RCF is 25.9-26.6 [8]. Only for time instant t_4 , where RCF is 25.87, constraint is violated. A colour coding (gray scale) is applied to the states of the cells and is given in Table (3).

Table 3 Colour-State-Value Pairs of CA

Colour	State	Value
White	Safe	1
Black	Unsafe	-1
Ash	Toggling	0

The state of the system was predicted from the values of t_0 to t_{24} based using 1-DCA.

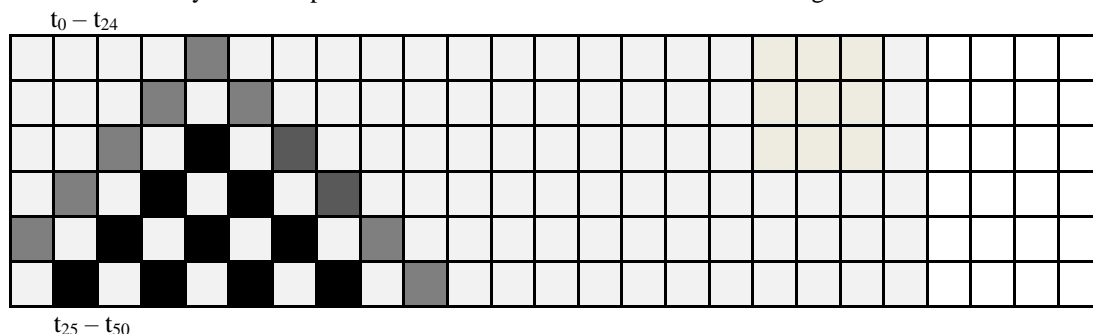


Figure 2: State prediction using 1-DCA

The first row shows the Ground State and subsequent rows depict the generations from Generation 1 to Generation 5. Since the percentage of unsafe states is increasing through generations, the system is predicted to move to unsafe state.

A comparison between the actual output obtained from the plant site in the next second, which is $t_{25} - t_{50}$ time instances and the results predicted using 1-DCA is given in Table (4).

Table 4 Comparison of prediction results using 1-DCA

RCF	Predicted State	Actual State
26.1	1	1
26.15	-1	1
26.11	1	1
26.12	-1	1
26.23	1	1
26.07	-1	1
26.05	1	1
26.11	-1	1
26.14	1	1
25.99	0	1
26.22	1	1
26.14	1	1
26.12	1	1
26.12	1	1
25.84	1	0
26.11	1	1
26.14	1	1
26.15	1	1
26.06	1	1
26.25	1	1
26.10	1	1
25.92	1	1
26.14	1	1
26.14	1	1
26.13	1	1

The predicted values are compared with the real plant site alarm cases, to analyse the accuracy of the results obtained through prediction. In the real thermal power plant site, the plant data are captured once in every 40 ms, which means 25 instance values are obtained in every second. It is observed in the site that, if the values of parameters are constantly increasing or decreasing, then the values cross the optimal limits, which results in an alarm case.

The value 0 in the predicted output denotes ‘toggling state’, which means the state can either move to safe state or unsafe state in the next generation. The conclusions drawn from Table (4) are:-

- Total number of mismatches = 5
- Total number of predictions = 25
- Percentage of mismatches = 0.2
- Percentage of matches = 0.8

Since the percentage of matches is only 80%, to make prediction more accurate, we have experimented with two parameters of pulveriser for prediction of states.

4.2.2 Results on prediction using 2-DCA

Since RCF and millcurrent (MC) are two dependent attributes, we have implemented two-dimensional CA with these parameters. These prediction results are updated to the rule base of the inference engine which makes it a control system with feedback. Data captured from the plant site in 1 second for the dependent attributes, namely RCF and MC is shown in Table (5), which gives 25 instance values.

Table 5* Domain values of two dependent attributes of pulveriser at various time instances

<i>Time Instant</i>	<i>Raw coal flow</i>	<i>Mill current</i>	<i>Time Instant</i>	<i>Raw coal flow</i>	<i>Mill current</i>
<i>T</i>	25.90	43.54	<i>t+13</i>	45.42	33.98
<i>t+1</i>	26.55	34.50	<i>t+14</i>	25.93	34.53
<i>t+2</i>	26.50	33.94	<i>t+15</i>	26.43	33.93
<i>t+3</i>	44.50	43.45	<i>t+16</i>	25.94	33.95
<i>t+4</i>	44.50	33.95	<i>t+17</i>	45.47	34.55
<i>t+5</i>	45.35	33.94	<i>t+18</i>	25.99	33.94
<i>t+6</i>	26.00	34.00	<i>t+19</i>	46.65	33.98
<i>t+7</i>	46.55	34.05	<i>t+20</i>	25.95	33.98
<i>t+8</i>	45.00	46.06	<i>t+21</i>	45.00	52.00
<i>t+9</i>	25.93	46.05	<i>t+22</i>	25.94	52.00
<i>t+10</i>	25.93	33.93	<i>t+23</i>	55.00	52.00
<i>t+11</i>	36.00	44.00	<i>t+24</i>	45.44	34.00

*Data obtained from the plant site of TTPS

The constraint or the optimal value for rawcoalflow is **25.9-26.6** and the optimal value of millcurrent is **33.93-34.9**[8].

Table (5) is reduced to a 5×5 matrix, matrix A of three values $\{0, 1, -1\}$ using the three rules defined in the methodology. The gray-scale coding used is given in Table (3). The result is shown in Figure (3).

GroundState = Matrix A
 = [0, 1, 1, -1, 0;
 0, 1, 0, -1, 0;
 1, -1, 1, 0, 1;
 1, 1, 0, 1, 0;
 1, -1, 0, -1, 0];

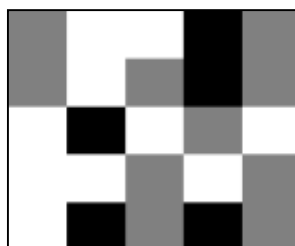


Figure 3 Ground State of 2-D CA

In this implementation, from the ground state, any number of generations can evolve. The number of iterations is limited by the percentage of unsafe states in a generation. If the percentage is beyond the permissible threshold value, prediction is made that the system is moving towards unsafe state. Here, the number of evolutions is limited to Generation 6, since the percentage of unsafe cells is increasing through generations. The prediction results obtained through generations from Generation 1 to Generation 6 is shown in Figure (3a) to Figure (3f).

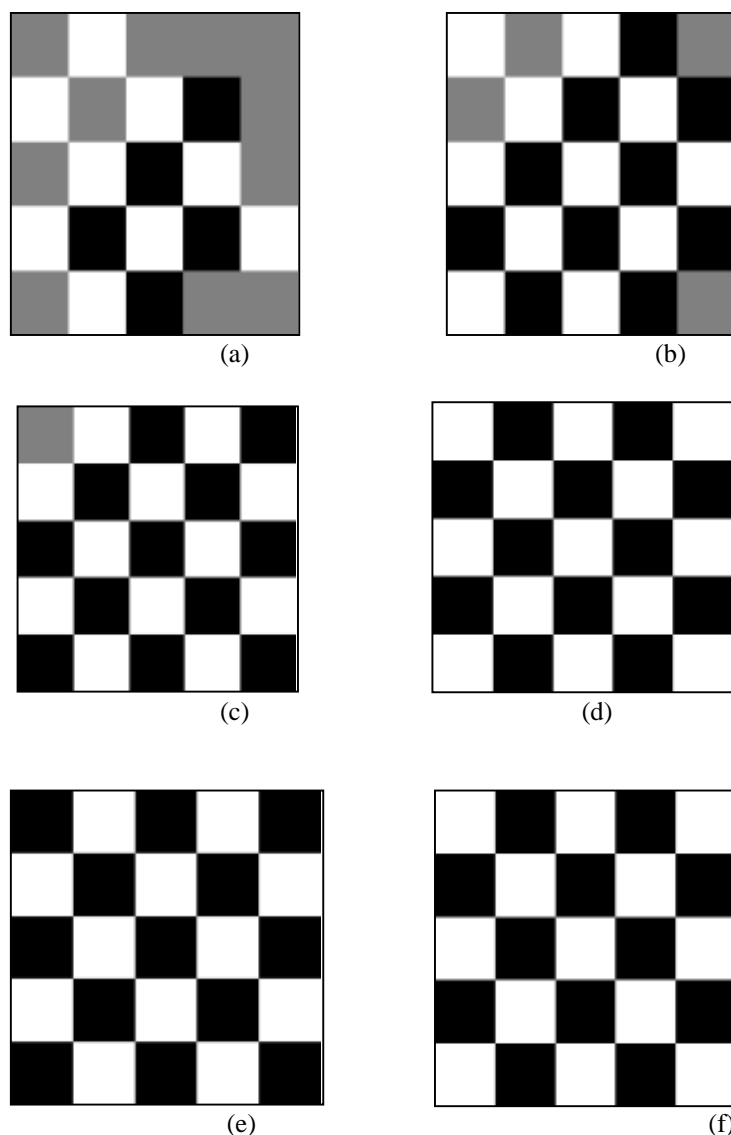


Figure 3 (a) Generation 1 (b) Generation 2 (c) Generation 3 (d) Generation 4 (e) Generation 5 (f) Generation 6

The above pattern continues for N generations, with unsafe and safe state in the cells. Hence, from the results, prediction can be made that the sub-system pulveriser can move to an unsafe state because of the values of the two dependent attributes of pulveriser, RCF and MC. As a result, the whole system of thermal power plant may move to unsafe state because of the misbehaviour of the component pulveriser or MillError.

4.3 Feedback Control System

The fault prediction system proposed here acts as a feedback system, which enables the rules to be updated based on the predictions made. Therefore, the system automatically updates with new rules in

a closed loop manner by routing back its output as inputs. This helps to track the system's operations and to identify conditions that are unsafe or potentially unsafe. The main advantage of the proposed system is that this could be used to update the rule-base automatically through a feedback to make the system more robust. The schematic diagram of **Co-MPc** with feedback is depicted in Figure (4).

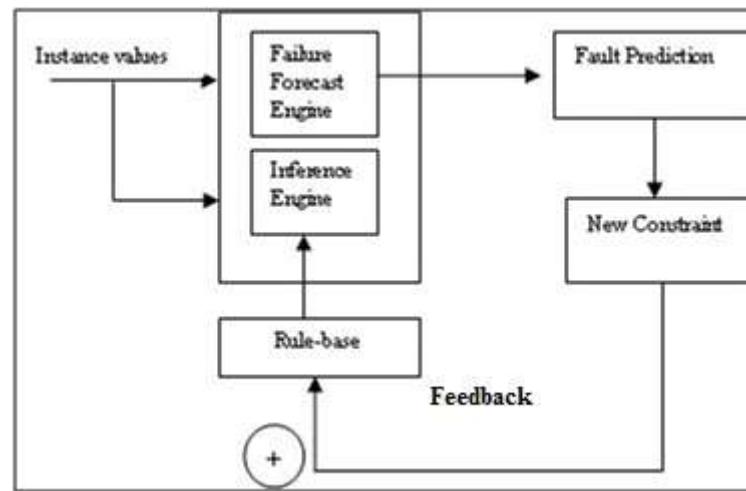


Figure 4: Predictive Model with feedback

In this work, the results obtained through prediction in the failure forecast engine is used to update the rule-base of the inference engine. This helps in creating a more precise and accurate rule-base which forms the foundation for future fault detections. The inference engine and the failure forecast engine of the model helps to validate the thermal power plant system by verification using on site data and predicting the behaviour or misbehaviour of the sub-system pulveriser in a dynamic manner, which would increase the performance and efficiency of the system.

V.RESULTS AND DISCUSSION

Various experiments were conducted to analyse the performance of failure detection through the Inference Engine. The inference engine works on the three categories of rules defined in the rule-base, namely domain specific constraints, inter-class constraints and dependent attributes constraints. If any of the constraints defined in the OCL system is violated, an alarm is generated.

The performance of failure prediction through the Failure Forecast Engine, which has been applied to TTPS using 1-DCA and 2-DCA, also has been analysed. When a single parameter was used for prediction using 1-DCA, the accuracy obtained was 80%. In order to increase the accuracy, two parameters were used for failure prediction, using 2-DCA. A comparison of the predicted output using 2-DCA and actual output obtained from plant site, for time instances t_{26} - t_{50} is given the Table (6).

Table 6 Comparison of predicted output and actual output for time instances $t_{26}..t_{50}$

<i>Actual input ($t_1..t_{25}$)</i>		<i>Ground state ($t_1..t_{25}$)</i>	<i>Predicted output(CA)</i>	<i>Actual Output from plant site</i>
<i>RCF</i>	<i>MC</i>		$t_{26}..t_{50}$	
26.02	35.19	0	1	1
26.03	34.77	1	1	1
26.18	34.74	1	1	1
26.16	34.56	1	1	1
26.07	34.58	1	0	1
26.16	34.67	1	0	1
26.13	34.81	1	1	1
26.10	34.84	1	1	1
26.27	35.37	0	1	1
26.27	35.37	0	1	0
26.13	34.95	0	1	1
26.21	34.41	1	1	1
26.15	34.92	0	1	1
26.35	35.01	0	0	1
26.11	34.42	1	-1	1
26.19	34.65	1	0	1
26.12	34.53	1	1	1
26.20	34.98	0	1	1
25.91	34.99	0	0	0
26.15	35.06	0	1	1
26.19	35.07	0	1	1
26.19	34.65	1	1	1
26.15	34.70	1	1	1
26.00	34.77	1	1	1
26.29	34.89	1	0	1

The predicted values are compared against the real plant site alarm cases, and it is observed in the site that, if the MC and RCF values are constantly increasing or decreasing, then the values cross the optimal

limits, which results in an alarm case. The value 0 in the predicted output denotes ‘toggling state’, which means the state can either move to a safe state or an unsafe state in the next generation. The conclusions drawn from Table (6) are:-

- Total number of mismatches = 2
- Total number of predictions = 25
- Percentage of mismatches = 0.08
- Percentage of matches = 0.92

Here, when 2-DCA was used, the percentage of matches increased to 92%. The prediction using cellular automata also reveals that, the presence of a toggling state or an unsafe state in one generation results in an unsafe state after n generations. Also, if there is an unsafe state in a generation, it is getting increased in the next generations. These results show the accuracy of the prediction model.

Results are presented in terms of Accuracy, Sensitivity or True Positive Rate (TPR), Specificity or True Negative Rate (TNR) and F-Measure in Table (7).

Table.7 Comparison of 1-DCA and 2-DCA on quantitative measures

Measures	1D CA	2D CA
TruePositive (TP)	20	22
TrueNegative (TN)	0	1
FalsePositive (FP)	1	1
FalseNegative (FN)	4	1
Accuracy	80%	92%
Sensitivity(TPR)	0.83	0.95
Specificity(TNR)	0	0.5
Precision	95.2%	95.6%
Recall	83.3%	95.6%
Fmeasure	88.9%	95.6%

Since 2-DCA gave better prediction results, the accuracy of 2-DCA based prediction at different time intervals($t_0..t_5$) are summarized in Table (8). Receiver-operating characteristic (ROC) curves are an excellent way to evaluate the results. ROC curves based on the predictions at different time intervals is shown in Figure (5).

Table 8 Accuracy of prediction at different time instances

Time instance	1-Specificity	Sensitivity(TPR)
time= t_0	0.02	0.4
time= t_1	0.04	0.5
time= t_2	0.06	0.6
time= t_3	0.08	0.83
time= t_4	0.13	0.97
time= t_5	0.5	1

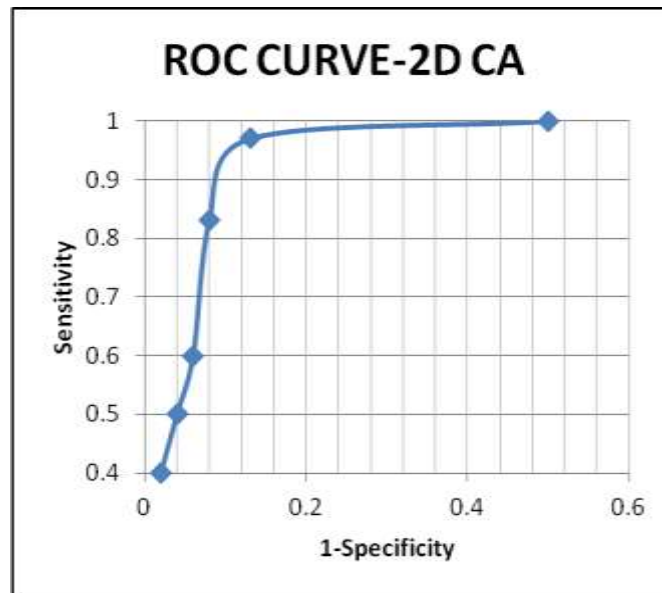


Figure
5.5 ROC curve of 2-DCA prediction results

The area under curve of the ROC reveals that the prediction results are accurate. Thus the findings suggest significantly improved forecasting performance.

VI.CONCLUSION

An object model for validating the functionality and failure forecasting of a complex constrained real-time system has been presented, with a rule-based logical inference engine and a cellular automata-based failure forecast engine, in this paper. Both 1-DCA and 2-DCA were employed for prediction of the state of the system and it is found that 2D cellular automata generate prediction results with an accuracy of 92%. The error can be reduced by optimising the CA rules thereby creating a robust failure forecast engine. The system also allows updating the rules of inference engine based on the forecast, thus guarantees to know the behaviour of the system. The pulveriser sub-system of the thermal power plant system is taken to demonstrate the model and has been applied on it successfully. The result of validation and prediction of the state of the complex system through the Co-MPc is in accordance with the actual values in the real thermal power plant. Thus, the model is well suited for fault prediction in similar complex systems.

REFERENCES

- [1]Chen, L., Fan, G. and Liu, Y. (2016a) "A formal method to model and analyse QoS-aware fault tolerant service composition", International Journal of Computational Science and Engineering, Vol. 12, Nos. 2/3, pp.133–145.
- [2] Chen, L., Fan, G., Zhang, H. and Xiao, L. (2016) "Petri nets-based method to model and analyse the self- healing web service composition", International Journal of High Performance Computing and Networking, Vol. 9, Nos. 1/2, pp.8–18.
- [3]Dresden OCL Toolkit, Dresden University of Technology [online] <http://dresden-ocl.sourceforge.net/index.html>. (Accessed : March 10 2014).
- [4] Karafyllidis, I. and Thanailakis, A. "A model for predicting forest fire spreading using cellular automata", Ecological Modeling, Vol. 99, No.1, pp.87–97.

- [5] Lo, S-C. and Hsu, C-H. "Cellular automata simulation for traffic flow with advanced control vehicles", in Computational Science and Engineering Workshops, 2008, CSEWORKSHOPS '08, 11th IEEE International Conference on IEEE pp.328–333.
- [6] OMG Unified Modeling Language Specification, v1.5. Mar. 2003. Available as OMG document formal/03-03-01.
- [7] Palma-Orozco, R., Palma-Orozco, G., De Jesus Medel-Juarez, J. and Jimenez-Benitez, J.A. "An approach to centralized control systems based on cellular automata", in Computational Intelligence in Security for Information Systems, in the series Advances in Intelligent and Soft Computing, Vol. 63, pp.187–191. Springer Berlin Heidelberg.
- [8] Singh, B.R., Valsalam, S.R., Pratheesh, H., Sujimon, K.T. and Aditi, C. "Real time pulverised coal flow soft sensor for thermal power plants using evolutionary computation techniques", ICTACT Journal on Soft Computing, January, Vol. 5, No. 2, pp.911–916.
- [9] Valsalam, S.R., Anish, S. and Singh, B.R. "Boiler modeling and optimal control of steam temperature in thermal power plants", Journal of Energy and Power Engineering, Vol. 5, No 8, pp.677–684.
- [10] Wei, J-L. et al. "Development of a multi segment coal mill model using an evolutionary computation technique", IEEE Transactions On Energy Conversion, September, Vol. 22, No. 3, pp.718–727.
- [11] Zhang, Y.G., Wu, Q.H., Wang, J., Oluwande, G., Matts, D. and Zhou, X.X. "Coal mill modeling by machine learning based on onsite measurements", IEEE Transactions on Energy Conversion, December, Vol. 17, No. 4, pp.549–555.
- [12] Zhu, L. and Xu, S. (2015) "Prediction algorithm based on web mining for multimedia objects in next-generation digital Earth", International Journal of Embedded Systems, Vol. 7, No. 1, pp.79–87.
- [13] Sakieh, Yousef, Amiri, B. J, Afshin Danekar, Jahangir Feghhi Sadeq Dezhkam. "Simulating urban expansion and scenario prediction using a cellular automata urban growth model, SLEUTH, through a case study of Karaj City, Iran." *Journal of Housing and the Built Environment* 30.4 (2015): 591-611.
- [14] Sree PK, Babu IR, SSSN Usha Devi N "Cellular Automata in Splice Site Prediction". *MOJ Proteomics Bioinform* 1(2): 00013.
- [15] Isermann, Rolf. "Model-based fault-detection and diagnosis—status and applications." *Annual Reviews in control* 29.1 (2005): 71-85.
- [16] Patton, R. J., F. J. Uppal, and C. J. Lopez-Toribio. "Soft computing approaches to fault diagnosis for dynamic systems: a survey." *4th IFAC Symposium on Fault Detection supervision and Safety for Technical Processes*. 2000.
- [17] Jiang, Yue, Bojan Cukic, and Yan Ma. "Techniques for evaluating fault prediction models." *Empirical Software Engineering* 13.5 (2008): 561-595.
- [18] Itami, Robert M. "Simulating spatial dynamics: cellular automata theory." *Landscape and urban planning* 30.1-2 (1994): 27-47.



Landscape analysis for watershed based planning- a study on Peruvamba river basin of Kerala

Abhilash T.K¹, Dr. Jayapal G² and Dr.T.K.Prasad³

1. Research Scholar, Dept of Geography, Kannur University

2. Assistant Professor and Head, Dept of Geography, Kannur University

3. Assistant Professor and Head, Dept of Geography, Govt. College kariavattom, Thiruvananthapuram.

Abstract

Landscape analysis is a process of studying, describing and interpreting the landscape ecology of an area and generally with the goal of assessing the impact of human on that space. Landscape changes always take place at the micro level as their immediate causes are inherently local, at the same time the impact can be felt at macro level due to aggregate nature of the effects. Landscape of a region can be examined on the basis of various micro-level landforms. Present study is intended to examine the geomorphological setting of Peruvamba river basin of Kannur district in Kerala. The watershed has a total area of 293 sq km covering 19 Revenue villages. The study area is characterised with highly complex relief and drainage pattern. In order to analyse their morphometry a detailed analysis on regional landscape has been carried out. A detailed study of various land forms and processes in the Peruvamba River basin has been carried out as part of this study and the region is divided into six distinctive landform units.

Key words: 1.landscape analysis,2. watershed based planning, 3.drainage basin, 4.piedplain, 5.dissected plateau,6.landform units.

Introduction

Landscape analysis is a process of studying, describing and interpreting the landscape ecology of an area and generally with the goal of assessing the impact of human on that space. Resources patches and a landscape network of connecting corridors are identified, described and classified. Landscape analysis and landscape planning are closely related and overlapping practices. Analysis is a pre requisite step for any planning, but is ongoing as planning proceeds (Thakur et.al 2008). This paper analyses the landscape characteristics of the study area with special reference to its geomorphic parameters.

The most significant fact about the earth surface is that each area is unique and has its own assemblage of bio-physical setting. Since the combination of landforms and geomorphic



processes varies from one region to the other, a proper understanding of them is essential for planning and development (Prasad 2018). Landscape associates people and place. A natural landscape is made up of a collection of landforms, such as mountains, hills, plains, and plateaus. Lakes, streams, soils (such as sand or clay), and natural vegetation are other features of natural landscapes. A landscape that people have modified is called a cultural landscape. People and the plants they grow, the animals they care for, and the structures they build make up cultural landscapes. Such landscapes can vary greatly. Landscape changes always take place at the micro level as their immediate causes are inherently local, at the same time the impact can be felt at macro level due to aggregate nature of the effects. Present study is intended to examine the geomorphological setting of Peruvamba river basin of Kannur district in Kerala.

Study area

The Peruvamba watershed lies between 12° 0' to 12°15' North latitudes and 75°10' to 75°20' East longitudes located in Kannur and Kasargod districts of Kerala state (Fig.1). It is bounded by Taliparambataluk of Kannur district and Hosdurgtaluk of Kasargod district in north, Kannur and Taliparambataluk of Kannur district in south, Taliparambataluk of Kannur district in the east and Arabian Sea in the west. The watershed has a total area of 292.88 sq km covering 19 villages spread over 12 panchayats, 1 municipality, 3 blocks and 2 districts.

Objectives

To examine the physiographic setting of the region through landscape analysis and to study the form and processes of various landform units.

Methodology

The prime objectives of the study is to analyse the complex physical set up of Peruvamba river basin through landscape analysis. As the study area is characterized by diversified local relief, it is essential to classify it into different landform units. The relative relief, slope nature of topography etc. are considered. The Geo-platform of ISRO named *Bhuvan* provides thematic layers of geomorphological units at three levels. The study area is categorized into 6 landform units based on Bhuvan Level 2 classification. Form and processes in these units are analysed through field study and GIS techniques.

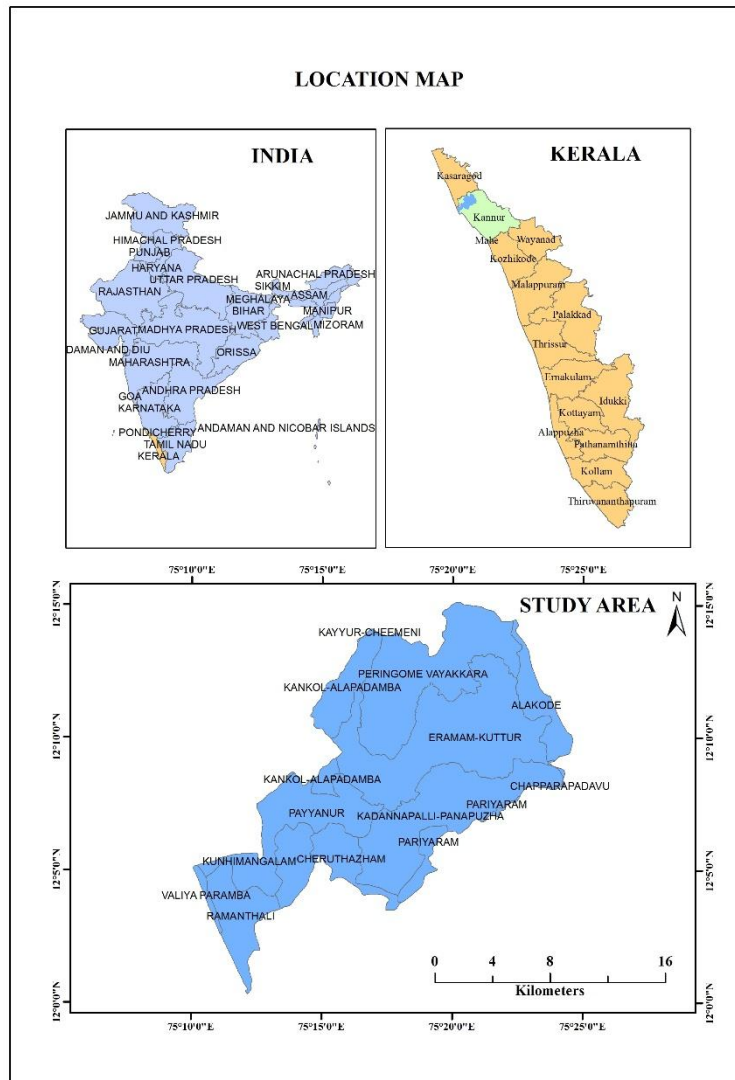


Fig 1

Results and discussion

The study area is endowed with a well-integrated system of drainage. The major river draining through this watershed is the Peruvamba River which has a length of 51 km. The River emerges from the densely forested hill slopes of the Western Ghats near *Pekunnu* at an altitude of 356 m above MSL. The head stream is known as Panappuzha and another head stream Kallankulamthode rises from Ezhilamvayal. Both the head streams joins at Korom and form the main stream of Peruvamba (Fig 2). The river is west flowing and its tributaries are Vannattipuzha, Mutalathodu, Mankarthodu, Chenattichal, Mukkottonkarachal, Aruvanchalthodu,

Challachal and Chankurichal. At Ezhimala, river bifurcates, while one branch joins the Kavvayi backwaters and other joins the Arabian Sea. The river joins Arabian Sea very near to Payyannur town. Most of the streams are perennial in nature. The drainage pattern appears to be dendritic.

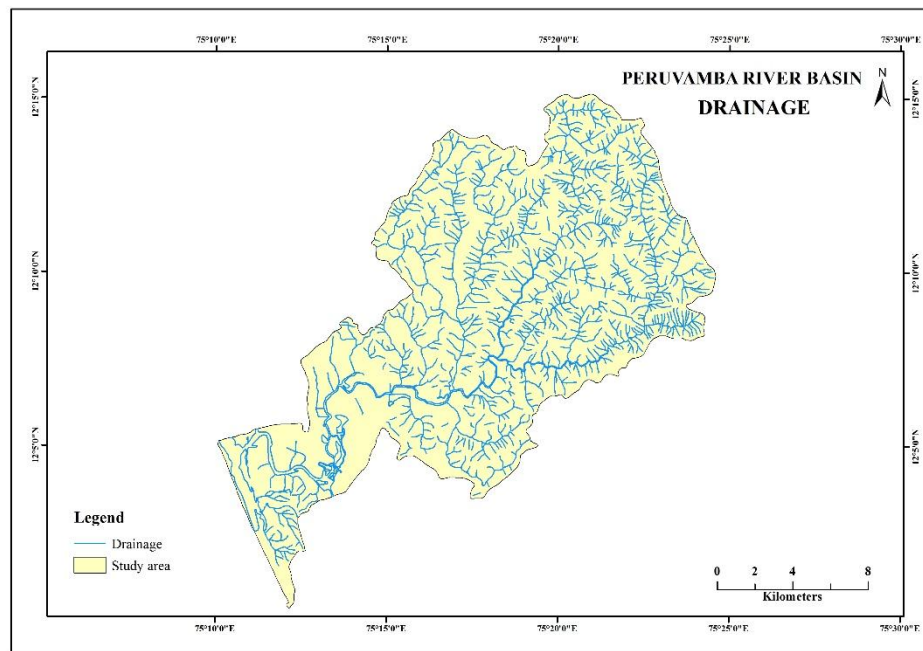


Fig 2

The major geological formation of the watershed is Archean (Soman 1996). The main lithological units are charnakite, calc granulite, coastal sand alluvium, anorthosite, biotite horn blende gneiss and sand stone and cly. Charnakite mainly found central part to north eastern part of the basin. Both the sides of charnakite region found calcgranuite. Coastal sand and alluvium constitute the next important lithological unit which is located in south eastern part of the basin. Small patches of laterite found in north western part and gabbro and granophyre seen in south of Peruvamba river basin. (Fig 3).

Soils of watershed vary in their texture, depth, slope, internal drainage and degree of erosion. The National Bureau of Soil Survey and Land use Planning has classified the soils of Kerala in to 38 soil units in association of two soils and numbered them serially from K01 to K38 based on characteristics like soil texture, surface gravelliness, soil reaction, slope, soil

erosion, depth of water table drainage etc. The salient attributes of the soils occurring in different physiographic regions of the watershed are furnished in Fig 4.

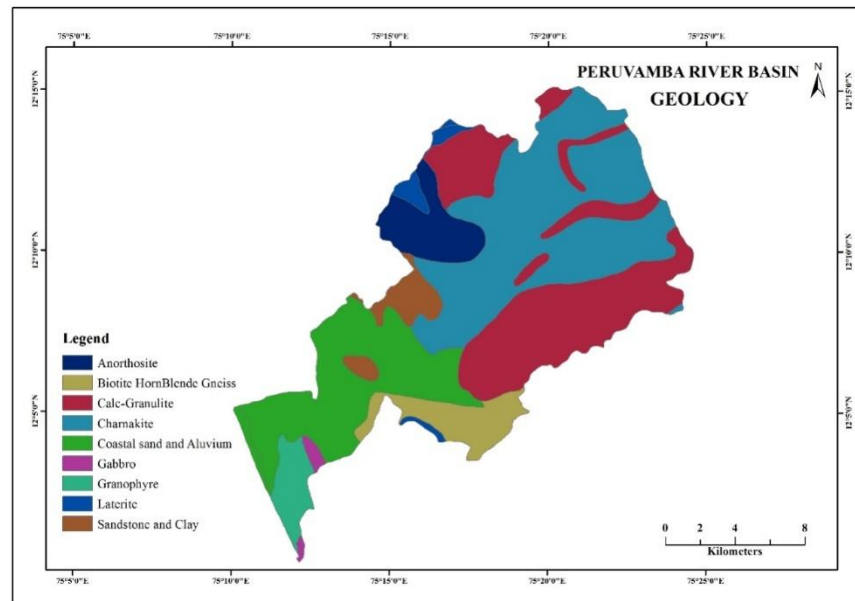


Fig 3

The major five type of soils are found in the study area. They are K01, K09, K10, K13, and K24. K10 soil category constitute major share of Peruvamba river basin and it seen throughout the highland and midland region. The next important category of soil is K13. It is mainly found in small patches the entire basin except lowland region and surrounded with K10 soil type. K24 category is a highland soil, which is found northern part of PeringomVayakkarapanchayath. K09 and K01 is mainly found in lowland areas of the basin. Major soils characteristics of Peruvamba river basin describes in table 1

Table 1 Peruvamba River Basin- Soil characteristics

Sl no	Map symbol	Depth	Texture	Slope	Drainage
1	K01	Very deep (vd)	Sandy (s)	Very gentle (vg)	Moderately well drained (mw)
2	K09	Very deep (vd)	Gravelly clay (gc)	Moderately steep (ms)	Well (w)
3	K10	Very deep (vd)	Gravelly clay (gc)	Gentle (g)	Well (w)
4	K13	Deep (d)	Gravelly clay (gc)	Gentle (g)	Well (w)
5	K24	Deep (d)	Gravelly loam (gl)	Moderately steep (ms)	Well (w)

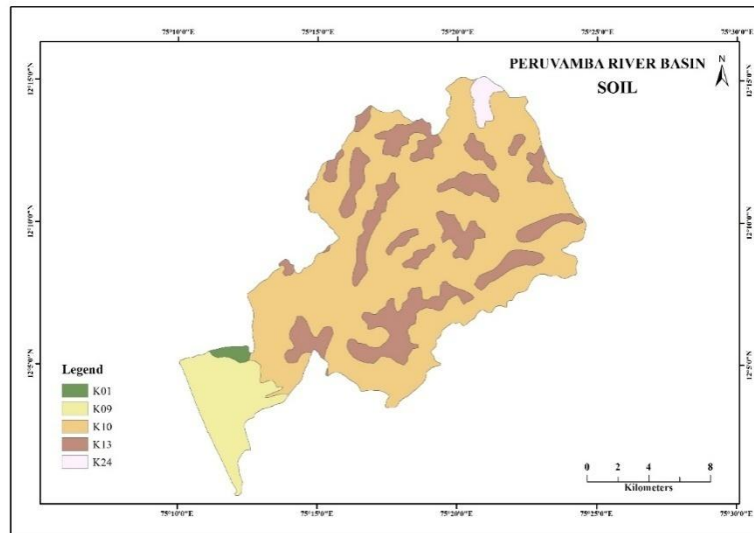


Fig 4

The traditional tripple fold classification (Highland, Midland and lowland) as fig 5 provides only a genral picture on physical setting of the region. The study area is charecterised with highly complex relief and drainage pattern. In order to analyses their morphometry a detailed analysis on regional landscape should be carried out. Landscepe of a region can be examined on the basis of various micro-level landforms. A detailed study of various land forms and processes in the Peruvamba River basin has been carried out as part of this study and the region is devided into six distinctive landform units. The following discussion is intended to provide an overview on these units.

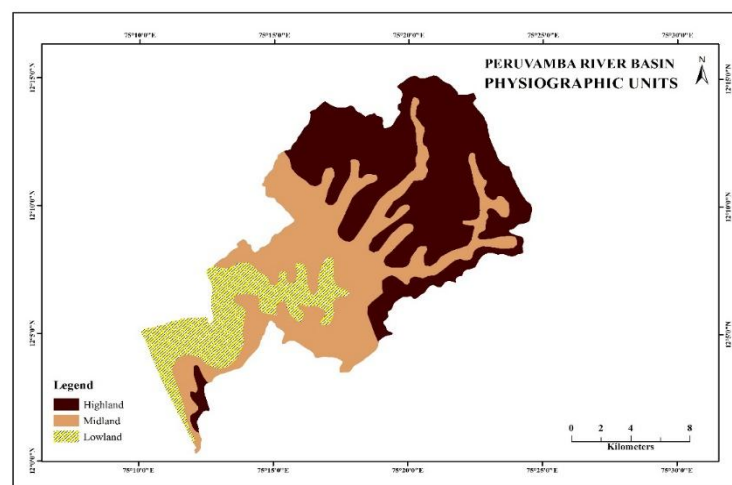


Fig 5

Landscape units

The study area is characterized with diversified local relief. The prime objective of this study is to analyse the complex physical set up of Peruvamba river basin through Micro-landscape analysis. The Geo-platform of ISRO named *Bhuvan* provides thematic layers of geomorphological units at three levels based on satellite data. The study area is categorized into 6 landform units based on Bhuvan Level 2 classification. Landform is an inorganic entity. Whereas inter play of nature as well as man has already enacted on this landform units. Hence these units can also be called as landscape units. The units identified area given in Table 2 and Fig6

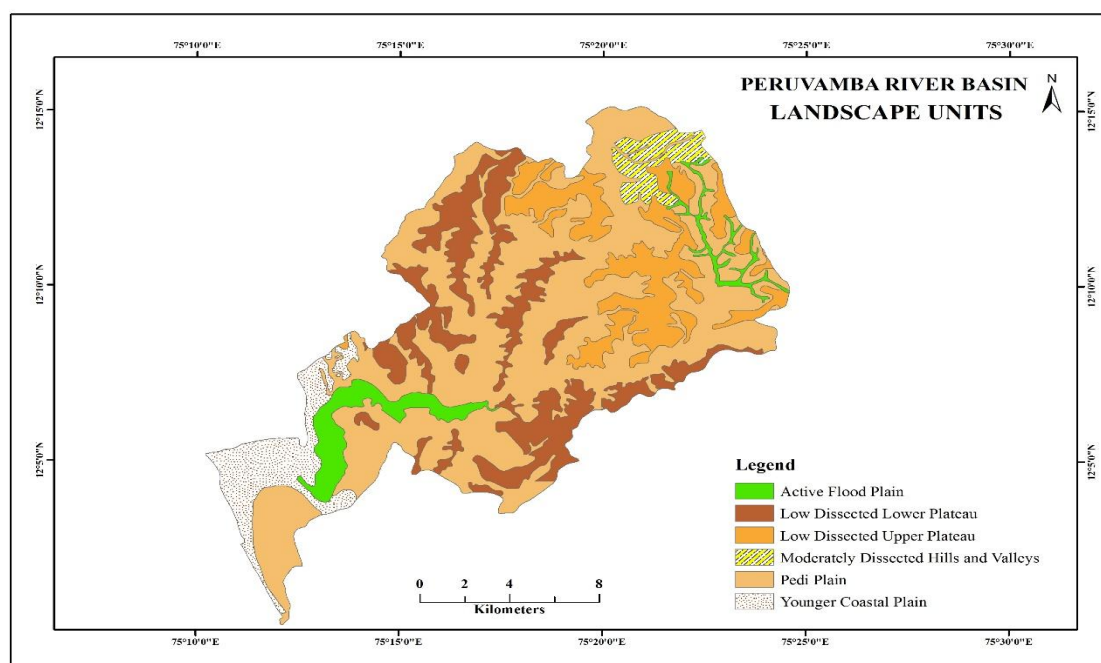


Fig 6

Table 2 Peruvamba River Basin- Landscape units (Compiled by the investigator)

Sl no	Landscape units	Area in Ha	Percentage
1	Active flood plain	1543.92	5.3
2	Low Dissected Lower Plateau	5010.70	17.1
3	Low Dissected Upper Plateau	3767.70	12.9
4	Moderately Dissected Hills and Valleys	886.21	3.0
5	Pedi Plain	16023.89	54.7
6	Younger Coastal Plain	2055.55	7.0
	Total	29287.97	100



A brief discussion on the salient features of geomorphological setting of these landform units is given below

1. Active flood plain

Active floodplain is defined as an area on either side of a stream/river which is regularly flooded on a periodic basis. It is inundated almost every year. By the end of the rainy season, it is marked with dry and braided channels rich in alluvium. These flood plains are often seen merging with meander plains with hardly any difference so that it is difficult to distinguish between meanders and cover flood plains. These plains are formed only because of variation in speed and decomposition. A flood plain, is a flat or nearly flat land adjacent to a stream or river that stretches from the banks of its channel to the base of the enclosing valley walls and experiences flooding during periods of high discharge. It includes the floodway, which consists of the stream channel and adjacent areas that carry flood flows, and the flood fringe, which are areas covered by the flood, but which do not experience a strong current. Floodplains are formed when a meander erodes sideways as it travels downstream. When a river breaks its banks, it leaves behind layers of alluvium (silt). These gradually build up to create the floor of the plain. Floodplains generally contain unconsolidated sediments, often extending below the bed of the stream. These are accumulations of sand, gravel, loam, silt, and/or clay, and are often important aquifers, the water drawn from them being pre-filtered compared to the water in the river. A flood plain consists of two parts. The first is the main channel of the river itself, called the floodway. Floodways can sometimes be seasonal, meaning the channel is dry for part of the year. Beyond the floodway is the flood fringe. The flood fringe extends from the outer banks of the floodway to the bluff lines of a river valley. Bluff lines, also called valley walls, mark the area where the valley floor begins to rise into bluffs. Flood plains are usually very fertile. It is composed of lower order landforms like Palaeo-channel, Abandoned Channel, Natural Levee, Oxbow Lake, Cut-off Meander, Meander scar, Crevasse Splay, Point Bar, Braid Bar, Lateral Bar, Channel Bar, Channel Island, Valley Fill, Back Swamp and Flood Basin.

Geomorphology of active flood plain of Peruvamba River basin characterized with flood plain, riverine islands, lateritic plateaus, paleo ridges, beach complex and river channels. The published geological reports indicate that this unit constitutes calc granulite, charnakite, coastal



sand and alluvium, sand stone and clay deposits. The soils of these region exhibits textural characteristics typical of gravely clay and well drained soils. The of the K10 and K13 soils are deep to very deep. The slope of this area gentle to moderately steep and height ranges from 10 to 200 m above MSL. The underground water level of this zone is moderate to very good.

2. Low dissected Lower Plateau

A plateau that is eroded and broken into numerous smaller pieces becomes a dissected plateau. Dissected plateau is characterized with sharp relief features caused by severe erosion. The dissected plateaus are distinguishable from orogenic mountains by lack of folding, faults, metamorphism or magmatic activity. It is denudational in origin. Weathering, erosion, and denudation of the flat land results in the formation of dissected plateaus. Elevated flat uplands occupying fairly large area and bounded by escarpments/steep slopes on all sides. While standing on a dissected plateau, it is evident that, the mountains are almost the same level in height, are almost the same level in height, which indicates the previous level of plateau prior to origin. The minor landforms in this category include Plateau Top, Plateau Remnant, Plateau Mesa, Plateau Butte, Plateau Scarp, Plateau Valley, Plateau Bench and Plateau Residual Capping.

The units have gentle to steep slope surface and are mostly under lateritic lower plateau. The geologic structure of low dissected lower plateau mainly composed with Anthrosite, biotite horn blende gneniss, calc granulite, charnakite, coastal sand and alluvium, laterite sand stone and clay. The ground water level of this region is moderate to very good. The general altitude of this unit ranges from 20 m to 200 m above MSL. The depth of K10 and K13 soils is deep to very deep and texture consists of gravely clay, well drained one.

3. Low dissected Upper Plateau

It is structural in origin. The physiological properties are almost same in the category 2. The upper portion is denuded vigorously. Plateau Top, Plateau Remnant, Plateau Mesa, Plateau Butte, Plateau Scarp, Plateau Valley, Plateau Bench and Plateau Residual Capping comprise this unit.



The surface of low dissected upper plateau is characterized by relief ranges from 20 m to 200 m with slopes varying from gentle to steep slopes. The ground water level of this region also moderate to very good like low dissected lower plateau. The soils types noticed in this unit are K10 and K13 category with deep to very deep depth. These well drained soils texture are gravely clay. Calc granulite, coastal sand and alluvium, sand stone and clay, are identified geological structure of this region. These region are geomorphologically composed with lateritic lower plateau and river channel valley.

4. Moderately Dissected Hills and valleys

The rocks of this geomorphic subunit are moderately dissected as compared to those of highly dissected denudational structural hills and have witnessed moderate degree of weathering and drainage dissection. As compared to highly dissected denude-structural hills, the structural ridges are more pronounced in moderately dissected denudational structural hills due to the lesser degree of dissection and denudation. Major landforms here include Hogback, Cuesta, Strike Ridge, Homocline, Monocline, Antiformal Hill, Antiformal Valley, Synformal Hill, Synformal Valley, Intermontane Valley, Rift Valley, Dyke / Sill Ridge, Dome, Valleys, etc..

The moderately dissected hills and valleys of the Peruvamba River basin is that portion of the basin where the altitude ranges between 40 m to 200 m above MSL. This unit is characterized by a land area with gentle to steep slopes. The ground water status of this zone is moderate to very good. The major soils found in this unit are K10, K13 and K24 and these soils characterized with gravely clay and gravely loam, depth of these well drained soils is deep to very deep. This unit is also characterized by the presence of occurrence of numerous calc granulite and charnakite deposits. These region mainly composed with lateritic lower plateau and river channel valley.

5. Pediplain

A pediplain is an extensive flat terrain formed by the coalescence of pediments. The term is used in geology and geomorphology, and it is derived from the Latin words *pes*, which means "genitive case," and *pedis*, which means "foot." Pediment, on the other hand, is a gently sloping bedrock surface created by lateral erosion or by mechanical weathering. The process through



which pediplains are formed is called pediplanation, and the concepts that try to explain this phenomenon were first developed in 1942 by geologist Lester Charles King.

The formation of a pediplain relies on erosion, which is the force behind the creation of a pediment. The formation of a pediment has not been well documented, and accordingly remains a subject of study, but there are existing theories that attempt to explain the process. As water and wind slowly erode and disintegrate rock surfaces, they reduce mountain ranges into a series of pediments at the base, and these pediments gently slope outward, where they coalesce with each other to form one large plain, which is the pediplain. Typical pediments have slopes with angles between 0.5 and 7 degrees, and are concave in shape. Pediments are best formed in arid and semi-arid areas where rainfall is intense for brief moments of time. Pediments that form in humid areas are usually obscured by vegetation and may be hard to notice. A pediplain consists mostly of thin alluvial surfaces that have undergone extreme erosion, and therefore it is not compacted. It is believed that the pediplain could be the last stage in the evolution of landform, and the end result of erosion process. It comprises Residual Mound, Tor, Pediment, Rolling Plain, Wash Plain, Valley Fill, Gullied Land, Complex Upland (Lateritic), etc.

Lithologically the unit consists of anthrositebiotite horn blende, gueniss, calc granulate, charnakite, coastal sand and alluvium, gabbro, granophyre, laterite, sand stone and clay deposits. The ground water level of this landform is poor to very good. The general altitude ranges from 20 m to 200 m above MSL. The slope of the surface is gentle to steep and geomorphological characteristics of this region are flood plain, kayals/estuaries, paleo beach ridge, residual hills, river channel, swale valley, younger coastal plain. These area composed with K09, K10, K13 and K24 category of soils. Depth of the soils is deep to very deep and texture consists of gravely clay and gravely loam.

6. Younger coastal plain

A coastal plain is flat, low-lying land adjacent to a sea coast. Generally, the land surface of the Coastal Plain rises only about 30 meters above sea level and dips less than one degree. The primary reason for the flat nature of the Coastal Plain is the unconsolidated sediments that are the "bedrock" of the region. The sediments in the Coastal Plain have not been significantly compacted or cemented and have certainly not become rock. There is little resistance to erosion because of the nature of the sediments, and therefore ridges do not



form in the Coastal Plain. Cretaceous and Tertiary deposits dominating the western Coastal Plain show more relief than the younger Quaternary deposits that are close to the shoreline and constantly pounded by wave action and flooding. This unit is characterized with, Plain Beach, Beach Ridge, Swale, Tidal Flat, Inter Tidal Flat, Mud Flat, Tidal Inlet, Marine Terrace, Wave cut Terrace, Sea Cliff, Plain Spit, Longitudinal Bar, Barrier Bar, Offshore Bar, Lagoon, Mangrove swamp, Estuarine Island or marshy area.

The younger coastal plain of the Peruvamba River basin general altitude ranges from 20 m to 200 m above MSL. This unit located south western part of the drainage basin. The slope of the surface level gentle to moderately sloping. Coastal sand and alluvium, gabbro, granophyre, sand stone and clay deposits mainly found in younger coastal plain region. The ground water level is moderate to very good. This zone is composed with flood plain, kayals/ estuaries, paleo beach ridge, residual hills, river channel, swale valley, younger coastal plain. Mainly found K01, K09, K10 and K13 soils category. Depth of the soil is deep to very deep with well drained and moderately well drained soils. The texture of the soil is vary gravely clay and sandy.

Conclusion

Land resources being finite cells for judicious use to meet the ever-increasing demands. But Land is constantly under threat of degradation, mainly as a result of unsustainable and unplanned exploitation and mismanagement. It is now well realized that one of the major factors affecting sustainability is change in the pattern and level of terrain (SrikumarChattopadhyay 2001). Terrain analysis forms the basis for all sorts of planning towards sustainable development. The landscape analysis of Peruvamba river basin in Kerala underlines this fact.

References

1. Chattopadhyay, Srikumar, 1995, Terrain Analysis of Kerala- Concept, Methods and application; technical Monograph, STEC Government of Kerala, Thiruvananthapuram
2. Chattoadhyayet,al (2001) Landscape change and its Environmental and Human Dimensions, Selected Micro Level Studies under Different Biophysical Setting in Chalakudi Basin” (KRPLLD Project No 24/2000).
3. Prasad T.K. (2018) Land scape of Kannur, A geomorphological appraisal, IMPACT: International Journal of Research in Humanities, Arts and Literature (IMPACT: IJRHAL) ISSN (P): 2347-4564; ISSN (E): 2321-8878 Vol. 6, Issue 7, Jul 2018, 355-370



4. Prasad T. K. And Parthasarathy G.R, 2018, Laterite and laterization- A Geomorphological Review", International Journal of Science and Research (IJSR), Volume 7 Issue 4, April 2018, 578-583.
5. Surjith Singh Saini and Dr Gupta M P (2009) Relief analysis of Kaushalaya river watershed using Remote Sensing and GIS techniques, Panjab geographer, vol 5, October.
6. Soman K, 2004, Geology of Kerala, Geol. Soc. India, Bangalore, pp. 6-29.
7. Thakur B R, Praveen Kumar Thakur, Hari Prasad V and Agarwal S P (2008) Relief analysis of Sollani watershed using Remote Sensing and GIS technology, Panjab geographer, vol 4, October, pp. 26-36



Functional Behaviour of Urban Centers: A Geographical Analysis of Kannur District, Kerala

Shimod K P¹, Dr. Jayapal G² and Dr.T.K.Prasad³

1. Research Scholar, Dept of Geography, Kannur University

2. Associate Professor and Head, Dept of Geography, Kannur University

3. Assistant Professor and Head, Dept of Geography, Govt. College Kariavattom, Thiruvananthapuram.

Abstract

Urbanization is an index of transformation from traditional rural economies to modern industrial one. It is progressive long-term process of concentration of population in urban areas. This is inevitable when pressure on land is high, agricultural income is low, and population increase is excessive. It should be recognized that urbanization is not a calamity but a necessity. Urbanization is a positive force and urban growth is an impetus to development. Both accelerate industrialization to some extent, they permit change in the social structure by raising the level of human aspiration, facilitate the provision of public services to a large sector of the population, and make possible increased economic opportunities and improve living conditions for those people who remains in the rural areas. The present study examines the functional characteristics of the blocks in the study area. Composite functional index method is used to find out their rank. GIS platform used to compile the output and to generate maps according to the findings of the study.

Key Words : 1.Urbanization, 2.Functions, 3.GIS, 4.Facilities, 5.Hierarchy.

Introduction

Urbanization is an index of transformation from traditional rural economies to modern industrial one. It is progressive long-term process of concentration of population in urban areas. It is the population migration from rural area to an urban area that results in its expansion at the cost of rural or natural land. In recent times, urbanization has been associated with industrialization and economic development. Census of India classify an area into urban if it has a minimum population of 5000, with a density of population at least 400 persons per sq.km and at least 75 percent of male working population engaged in non- agricultural activities.

Urbanization trend in the state of Kerala shows marked peculiarities. The main reason for urban population growth is the increase in the number of urban areas and also urbanization of the peripheral areas of the existing major urban centers. However unlike the other parts of the country the Urbanization in Kerala is not limited to the designated cities and towns. The difference between rural and urban agglomerations is very negligible so far as Kerala is concerned. The variation of urban and rural content of Kerala from 1951 to 2011 shows the urban content has reached 47.72% in 2011 from the value of 13.48% in 1951. The present urban and rural content of Kerala resembles to that of the world population.

The growth of urban population of the Kannur district as well as urban area is increasing. This is contrary to the general trend of rural areas of coastal belt which is gradually urbanized. According to the 2001 census Kannur district ranked first in the percentage of urban population (50.35). This has been pushed down to 4th rank in 2011 census with 65.05 percentage of urban population. As per 2011 Census Kannur district has 9 Blocks. The present study is based on this administrative division of the district.



Study area

Kannur district lies between $11^{\circ} 40'N$ to $12^{\circ} 48' N$ latitudes and $75^{\circ} 10'E$ to $75^{\circ} 57'E$ longitude. The district is bounded by the Western Ghats in the east (Coorg district of Karnataka State), Kozhikode and Wayanad District in the south, Lakshadweep Sea in the west and Kasaragod, the northernmost district of Kerala, in the north (Fig. 1.1) with an area of 2966 sq km. According to 2011 census Kannur District has 9 blocks namely Kannur, Thalassery, Payyanur, Thaliparamba, Irrity, Irrikur, Edakkad, Peravoor and Koothuparamba. Kannur district has 4 class II towns, 27 class III towns, 34 class IV towns, one class V and VI town as per 2011 census. No class one town in the district even though it has a variety of natural and agricultural resources. This increase in the number class towns from 37 to 67 from the last decade made several spatial and social changes in the district.

Objectives

1. To analyse the spatial pattern of the urban centers in the study area
2. To study the functional behavior of the study area

Database and methodology

The present study mainly consists of secondary data sources. Data regarding the different aspects of population have been collected from the census of India. Secondary data collected from various governmental and quasi governmental agencies. Analysis of data where done by using cartographic and statistical techniques. Generation of geo-database, analysis, decision making and representation is done using Arc GIS software.

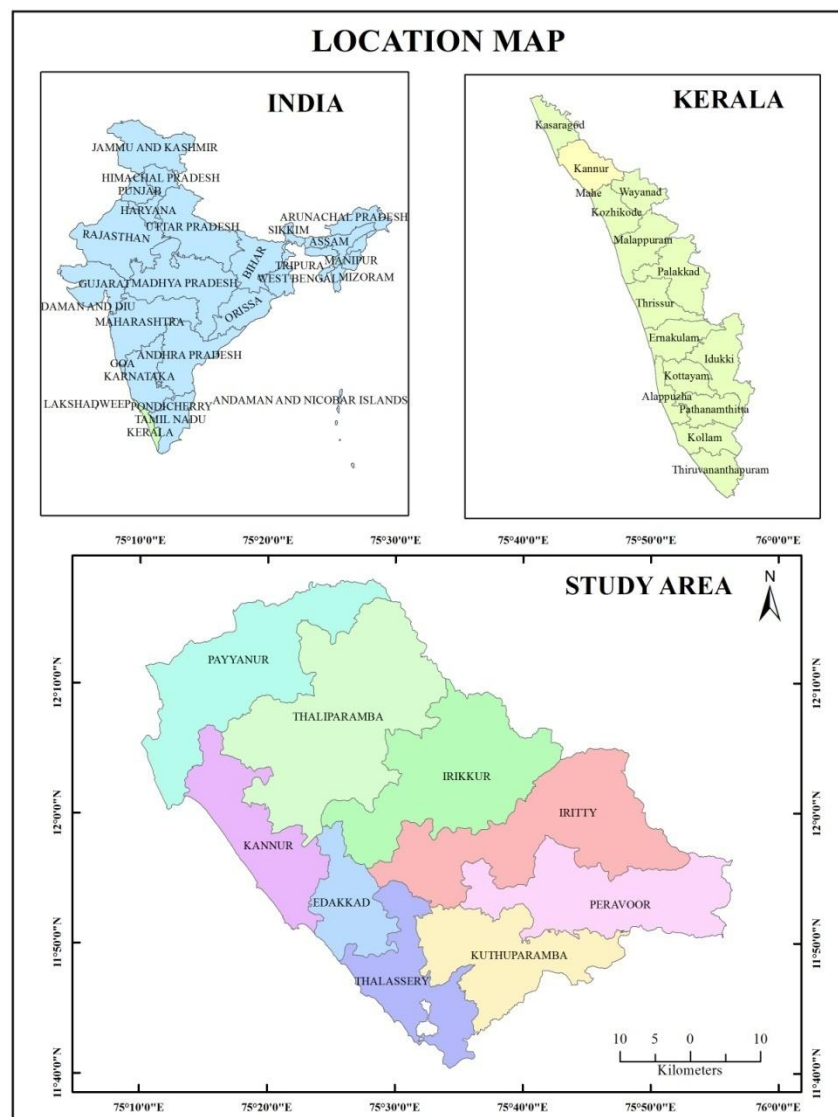


Fig. 1 – Location Map

Hierarchy Based on Facilities

Hierarchy of a town is an indication on the availability and hierarchy of various facilities in the area. It is a reflection on the size of population that depending town or the city for facilities. There are number of studies and different approaches and methods adopted by scholars to identify hierarchical arrangement of settlements. They are Scalogram technique by L. Guttman, population threshold and ranking of central places and functions by Berry and Garrison, Ranking of settlements on the basis of hierarchy of functions. This study describes the ranking the medium towns based on existing functional facilities in the district. Composite functional index method is used to find out their rank. Data from District Census Hand book 2011 and Panchayat Level Statistics for the year 2011 published by Economics and Statistics Department is used for the study.

The equation for this is as follows;



$$\text{Weightage of a facility} = \frac{\text{No. of facilities}}{\text{Population of the city}} \times 10000$$

This equation will give a scale-free data which can be easily used for the identification of composite index. The aggregate values of facilities of the towns yield the composite value. The calculation of which is done by the addition of all the weightages of facilities and dividing the values with the total number of facilities.

Facilities taken for hierarchy calculation are classified under five main heads they are:

Table 1.1 – Classification of Functional Facilities

Sl No	Functional Facilities	Functional Units
I	Communication	Post Office
		Telephone Exchange
		Railway Station
		Bus stand
		Boat Jetty/Ferry
		Airport
II	Education	LP & UP
		HS
		HSS & VHSS
		ITI/ITC
		Colleges
		Adult Education Center
		Anganavady
		Public Library
		Industrial training center
		Employment training center
III	Health	Hospitals
		Dispensaries
		Health center and Family welfare center
		Health clubs
IV	Recreation	Cinema Theatres
		Parks, Tourist centers
		Sports Club
		Arts Club
		Stadium
		Music/Dance Schools
V	Public Facilities	Market
		Reading Room



		Community Hall
		Handicraft centers
		Police Station
		Fire Station
		Banks (National/Scheduled/Cooperative bank)
		Public Comfort stations
		Public distribution centers
		Gas agencies
		Cremation center
VI	Industries	Wood processing
		Handloom
		Engineering Units
		Agro and food processing
		Building materials
		Paper and printing
		Miscellaneous

Source: Compiled by Researcher

Result and discussions

The rank of blocks has assigned based on the composite functional index method, in order to carry out the analysis first of all the facilities of blocks are identified and are given weightage for each facility. The weightage value of facilities is the indicator of their relative significance.

Transport and Communication Facilities

Transport and communication are a way for the people to overcome the barrier of physical distance. These facilities are important infrastructure for the overall economic development of any region. It is also known as the life line of any region because it affects the internal or external economic activities and the mobilization of goods and services. Post office, Telephone exchange, Bus Stand, Boat Jetty/Ferry, Railway station and Airport together contributed to the communication facilities. Even though all the blocks are well connected with the facilities like Bus Stand and Post Office the facilities like Railway Stations and Boat Jetty are confined to the coastal tracts of the district and are limited access to the high land areas. There is only one Airport in the district and which is in the Irritty block. Use of the mobile networks reduced the importance of telephone exchanges in almost all the cities in the district as well as in the state.

Education Facilities

Educational institutions are the places where people of different ages gain knowledge which includes playschools to the universities. These facilities in an area play a vital role in the development and living standard of the people living in that society. In a close observation within the education facilities it is clear that the basic educational facilities have got more importance with compared to the higher education facilities like HSS and Colleges. It is also necessary to give more importance to the higher order educational facilities otherwise the



population in these cities has to depend on other neighboring towns or the districts to avail the facilities to satisfy the population. In this study Educational Facilities includes the total number of LP & UP Schools, High Schools, Higher Secondary Schools & Vocational Higher Secondary Schools, ITI/ITC, Colleges, Adult Education Center, Anganavady, Public Library, Industrial training center and Employment training center.

Health Facilities

Health facilities indicate the place where the medical facilities are provided, which ranges from health center to the super specialty hospitals. The number and the quality of health facility in a region are one of the common measure in the quality and prosperity of the life. This facility includes Hospitals, Dispensaries, Health Centers and Family Welfare Center and Heath Clubs.

Recreation Facilities

This facility plays an essential part of the human life and they are in different forms shaped by the natural or individual interest by the surrounding population. Recreational activities can be communal or solitary, active or passive, outdoors or indoors, healthy or harmful, and useful for society or detrimental. A significant section of recreational activities are designated as hobbies which are activities done for pleasure on a regular basis. It is assumed that the higher quality of recreational facilities promote the regional development as well as the mental growth of the population residing in that area. A list of typical activities could be almost endless including most human activities. Here in this regard cinema theaters, parks, tourist club, sports club, arts club, stadium and music/dance schools are taken into account.



Table 1.2 – Block wise Total Number of Facilities

Sl No		Kannur	Thalassery	Payyanur	Thaliparamba	Irikkur	Edakkad	Iritty	Peravoor	Koothuparamba	Total
1	Post Office	27	53	74	66	33	32	38	26	60	409
2	Telephone Exchange	7	6	18	18	15	9	11	4	10	98
3	Boat jetty/Ferry	8	8	18	11	6	3	0	0	0	54
4	Bus Stand	5	2	10	7	4	0	4	2	14	48
5	Railway station	4	3	3	2	0	1	0	0	0	13
6	Air port	0	0	0	0	0	0	1	0	0	1
	Total	51	72	123	104	58	45	54	32	84	623
7	LP & UP	64	182	130	157	90	153	93	45	172	1086
8	HS	0	4	4	9	4	0	3	3	2	29
9	HSS & VHSS	37	22	47	34	24	19	19	8	24	234
10	ITI/ITC	6	4	7	3	5	8	5	2	4	44
11	Colleges	12	7	5	12	8	3	2	1	3	53
12	Adult Edu Center	0	43	6	26	9	15	16	4	23	142
13	Anganwady	174	238	318	218	159	153	165	86	161	1672
14	Pub. Lib	23	44	97	50	44	52	29	14	20	373
15	Industrial Training Center	2	4	5	4	2	4	3	1	3	28
16	Emplymnt Training Center	2	4	2	2	2	8	2	5	4	31
	Total	320	552	621	515	347	415	337	169	416	3692
17	Hospitals	44	23	26	32	9	17	22	3	4	180
18	Dispensaries	3	25	51	49	24	10	28	12	14	216
19	Helth C & Family WC	31	48	82	73	37	49	49	20	48	437
20	Health Club	18	7	14	12	3	11	5	3	8	81
	Total	96	103	173	166	73	87	104	38	74	914
21	Cenima Theatre	18	8	14	12	3	5	5	3	6	74

22	Parks, Tourist club	2	6	6	3	1	4	4	1	0	27
23	Sports Club	32	114	269	246	87	102	97	24	93	1064
24	Arts Club	10	72	214	306	41	48	95	12	65	863
25	Stadium	2	13	15	12	1	4	9	6	11	73
26	Music/ Dance School	8	15	11	10	4	8	4	6	9	75
	Total	72	228	529	589	137	171	214	52	184	2176
27	Market	12	12	9	6	3	8	5	3	3	61
28	Reading Room	52	135	99	183	85	104	81	16	56	811
29	Community Hall	6	17	38	16	7	11	8	3	12	118
30	Handicraft Cent	4	4	7	6	3	4	2	0	15	45
31	Police Station	4	5	3	5	5	2	3	2	4	33
32	Fire Station	1	1	1	1	1	1	2	1	1	10
33	Banks All	78	99	103	100	55	72	77	26	72	682
34	Public Comft Station	5	6	6	4	6	3	5	2	4	41
35	Public Dist Center	80	139	149	169	87	107	92	51	122	996
36	Gas Agencies	18	20	22	26	20	23	9	6	17	161
37	Cremation Center	20	6	113	63	18	12	5	8	6	251
	Total	280	444	550	579	290	347	289	118	312	3209
38	Wood Processing	32	8	2	0	0	2	0	0	0	44
39	Handloom	56	8	12	6	2	26	0	0	2	112
40	Engineering Units	73	6	12	12	6	37	3	1	8	158
41	Agro & Food	24	16	14	8	12	10	6	2	6	98
42	Building	9	7	4	6	3	2	2	1	6	40
43	Paper & Printing	16	9	2	0	0	7	0	0	2	36
44	Miscellaneous	86	28	14	15	18	46	12	4	24	247
	Total	296	82	60	47	41	130	23	8	48	735

Source: Compiled by Researcher



Industries

Industrial development had an important role in the economic development of any region. This will bring growth in the many other sectors such as transport and communication, educational facilities, banking facilities, public health facilities etc. This will lead to the development of living standard for the population residing in that area. In this study wood processing industries, handloom, engineering units, agro and food processing units, building, paper and printing and other small scale industries are taken into consideration.

Public Facilities

Public facilities are the services provided by the government to the public which includes the infrastructure, sanitation, transportation, health, reading rooms, markets, public distribution centers, emergency facilities like fire station, police station, banking institutions etc. The number and distribution of each of these facilities is an indicator of the regional development.

Ordering of blocks based on their infrastructure facility is very important factor in the regional planning contest. The analysis and the categorization of each facility in the town will give a clear outlook of the characteristics in a region in general and the town in particular. It will help the planners and government officials to identify the areas which face the shortage of certain facility or any group of facilities. It can also help the private developers and entrepreneurs to identify which infrastructure has facing more stress with the population size and which type of investment will be more accurate for the economic benefit of the proprietors and the collective development of the city environment.

Hierarchy of the block is derived by treating each town as a settlement and each block is given weightage considering the number of facilities present in the blocks and the population. Here the hierarchy of blocks have been determined on the basis of Composite Functional Index Method.



Table 1.3 – Block wise Functional Index of Facilities

Sl No		Kannur	Thalassery	Payyanur	Thaliparamba	Irikkur	Edakkad	Iritty	Peravoor	Koothuparamba	Total
1	Post Office	1.20	1.67	1.95	1.55	1.38	1.15	1.63	1.91	2.10	14.52
2	Telephone Exchange	0.31	0.19	0.47	0.42	0.63	0.32	0.47	0.29	0.35	3.46
3	Boat jetty/Ferry	0.35	0.25	0.47	0.26	0.25	0.11	0.00	0.00	0.00	1.70
4	Bus Stand	0.22	0.06	0.26	0.16	0.17	0.00	0.17	0.15	0.49	1.69
5	Railway station	0.18	0.09	0.08	0.05	0.00	0.04	0.00	0.00	0.00	0.43
6	Air port	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.04
	Aggregate Score	2.26	2.26	3.24	2.44	2.42	1.61	2.32	2.35	2.94	21.85
7	LP & UP	2.84	5.72	3.43	3.68	3.76	5.49	4.00	3.30	6.01	38.22
8	HS	0.00	0.13	0.11	0.21	0.17	0.00	0.13	0.22	0.07	1.03
9	HSS & VHSS	1.64	0.69	1.24	0.80	1.00	0.68	0.82	0.59	0.84	8.29
10	ITI/ITC	0.27	0.13	0.18	0.07	0.21	0.29	0.21	0.15	0.14	1.64
11	Colleges	0.53	0.22	0.13	0.28	0.33	0.11	0.09	0.07	0.10	1.87
12	Adult Edu Center	0.00	1.35	0.16	0.61	0.38	0.54	0.69	0.29	0.80	4.82
13	Anganwady	7.72	7.48	8.39	5.11	6.64	5.49	7.09	6.31	5.63	59.84
14	Pub. Lib	1.02	1.38	2.56	1.17	1.84	1.87	1.25	1.03	0.70	12.81
15	Industrial Training Center	0.09	0.13	0.13	0.09	0.08	0.14	0.13	0.07	0.10	0.97
16	Employment Training Center	0.09	0.13	0.05	0.05	0.08	0.29	0.09	0.37	0.14	1.28
	Aggregate Score	14.19	17.35	16.37	12.07	14.48	14.89	14.48	12.40	14.55	130.78
17	Hospitals	1.95	0.72	0.69	0.75	0.38	0.61	0.95	0.22	0.14	6.40
18	Dispensaries	0.13	0.79	1.34	1.15	1.00	0.36	1.20	0.88	0.49	7.35
19	Health & Family Welfare Center	1.37	1.51	2.16	1.71	1.54	1.76	2.11	1.47	1.68	15.31
20	Health Club	0.80	0.22	0.37	0.28	0.13	0.39	0.21	0.22	0.28	2.90
	Aggregate Score	4.26	3.24	4.56	3.89	3.05	3.12	4.47	2.79	2.59	31.96



21	Cinema Theatre	0.80	0.25	0.37	0.28	0.13	0.18	0.21	0.22	0.21	2.65
22	Parks, Tourist club	0.09	0.19	0.16	0.07	0.04	0.14	0.17	0.07	0.00	0.94
23	Sports Club	1.42	3.58	7.09	5.77	3.63	3.66	4.17	1.76	3.25	34.33
24	Arts Club	0.44	2.26	5.64	7.17	1.71	1.72	4.08	0.88	2.27	26.19
25	Stadium	0.09	0.41	0.40	0.28	0.04	0.14	0.39	0.44	0.38	2.57
26	Music/ Dance School	0.35	0.47	0.29	0.23	0.17	0.29	0.17	0.44	0.31	2.73
	Aggregate Score	3.19	7.17	13.95	13.80	5.72	6.13	9.20	3.81	6.43	69.41
27	Market	0.53	0.38	0.24	0.14	0.13	0.29	0.21	0.22	0.10	2.24
28	Reading Room	2.31	4.24	2.61	4.29	3.55	3.73	3.48	1.17	1.96	27.34
29	Community Hall	0.27	0.53	1.00	0.37	0.29	0.39	0.34	0.22	0.42	3.85
30	Handicraft Cent	0.18	0.13	0.18	0.14	0.13	0.14	0.09	0.00	0.52	1.51
31	Police Station	0.18	0.16	0.08	0.12	0.21	0.07	0.13	0.15	0.14	1.23
32	Fire Station	0.04	0.03	0.03	0.02	0.04	0.04	0.09	0.07	0.03	0.40
33	Banks All	3.46	3.11	2.72	2.34	2.30	2.58	3.31	1.91	2.52	24.24
34	Public Comfort Station	0.22	0.19	0.16	0.09	0.25	0.11	0.21	0.15	0.14	1.52
35	Public Dist Center	3.55	4.37	3.93	3.96	3.63	3.84	3.95	3.74	4.27	35.24
36	Gas Agencies	0.80	0.63	0.58	0.61	0.83	0.82	0.39	0.44	0.59	5.70
37	Cremation Center	0.89	0.19	2.98	1.48	0.75	0.43	0.21	0.59	0.21	7.72
	Aggregate Score	12.42	13.96	14.50	13.57	12.10	12.45	12.42	8.66	10.91	110.98
38	Wood Processing	1.42	0.25	0.05	0.00	0.00	0.07	0.00	0.00	0.00	1.79
39	Handloom	2.48	0.25	0.32	0.14	0.08	0.93	0.00	0.00	0.07	4.28
40	Engineering Units	3.24	0.19	0.32	0.28	0.25	1.33	0.13	0.07	0.28	6.08
41	Agro & Food	1.06	0.50	0.37	0.19	0.50	0.36	0.26	0.15	0.21	3.60
42	Building	0.40	0.22	0.11	0.14	0.13	0.07	0.09	0.07	0.21	1.43
43	Paper & Printing	0.71	0.28	0.05	0.00	0.00	0.25	0.00	0.00	0.07	1.37
44	Miscellaneous	3.81	0.88	0.37	0.35	0.75	1.65	0.52	0.29	0.84	9.46
50	Aggregate Score	13.12	2.58	1.58	1.10	1.71	4.66	0.99	0.59	1.68	28.01

Source: Compiled by Researcher



Table 1.4 – Composite Functional Index of Facilities

Sl No	Blocks	Communication	Education	Health	Recreation	Industries	Public Facilities	CFI
1	Kannur	2.26	14.19	4.26	3.19	13.12	12.42	49.44
2	Thalassery	2.26	17.35	3.24	7.17	2.58	13.96	46.55
3	Payyanur	3.24	16.37	4.56	13.95	1.58	14.50	54.21
4	Thaliparamba	2.44	12.07	3.89	13.80	1.10	13.57	46.87
5	Irikkur	2.42	14.48	3.05	5.72	1.71	12.10	39.48
6	Edakkad	1.61	14.89	3.12	6.13	4.66	12.45	42.86
7	Iritty	2.32	14.48	4.47	9.20	0.99	12.42	43.88
8	Peravoor	2.35	12.40	2.79	3.81	0.59	8.66	30.60
9	Koothuparamba	2.94	14.55	2.59	6.43	1.68	10.91	39.09
Avg. Score		2.43	14.53	3.55	7.71	3.11	12.33	43.67
Total		21.85	130.78	31.96	69.41	28.02	110.98	392.99

Source: Compiled by Researcher

Significance of Facilities

The weightage value of facilities is the indicator of their relative significance. Among the individual facilities the highest value is attained by the facility like Anganavady (59.84), LP and UP Schools (38.22), Public Distribution (35.24), Reading Room (27.34) and Banks (24.24) respectively. This mainly comes under the categories of Educational facilities and Public facilities. It indicates that in any society education has its own importance in the development of a region. While the lowest weightage values obtained by facilities like Fire station (0.40), Railway Station (0.43), Industrial training center (0.97) and Police station (1.23). These values show their relative significance in contributing the physical wellbeing and the quality of life in the study area. This doesn't mean that there is no importance for these kinds of emergency facilities in the development of any administrative units but it shows the lack of availability of these facilities to serve the general public. So the number of these facilities should be increased to address the needs of public.

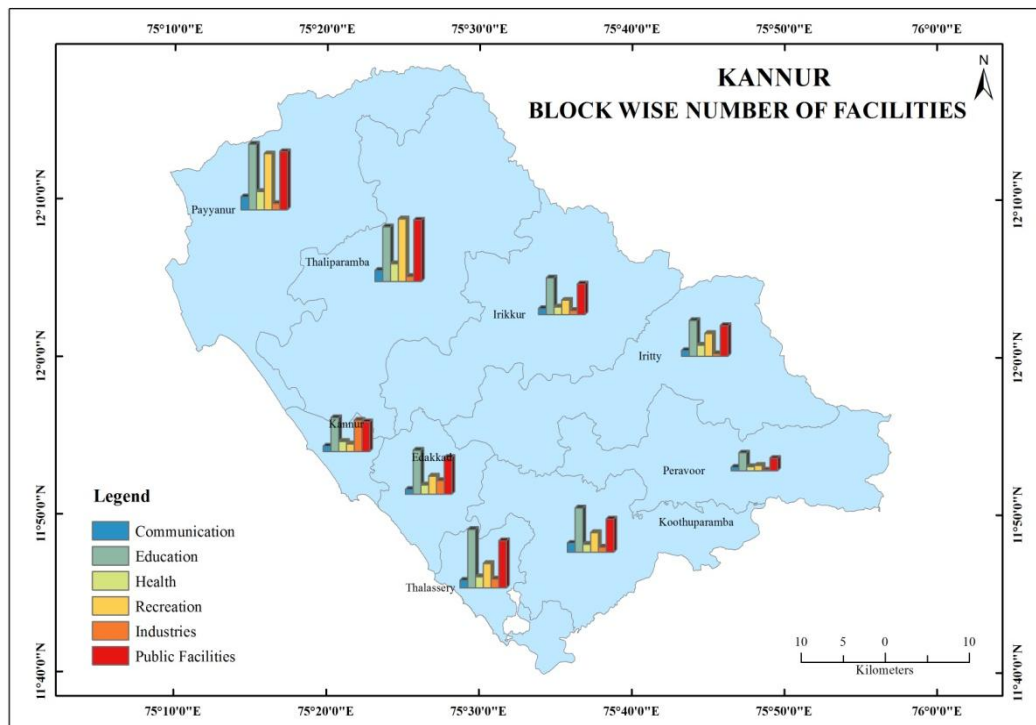


Fig. 2 – Block wise Number of Facilities

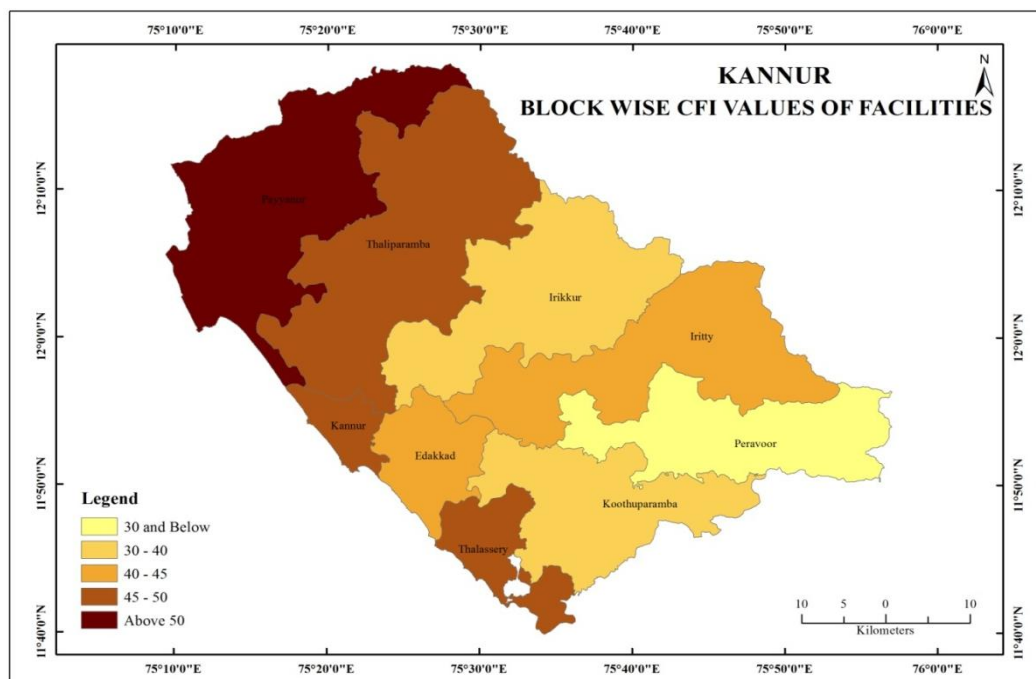


Fig. 3 – Block wise Cumulative Functional Index Value of Facilities



Conclusion

Kannur possesses an extensive urban development with more than 60% of the total population living in urban areas as per 2011 census. The district has been one of the most urbanized districts in the State even from 1991. The district was ranked 1st in the urban population in 1991 and 2001, later in 2011 census it changed to 4th rank. This shows that how far the facilities in the district played a vital role in the urban development in the district. The towns with comparatively higher number of functional infrastructure and the lower population got the higher ranking and vice versa. Among the facility categories used in this study the educational facility infrastructure contributes more towards a better urban environmental conditions and the quality life of urban dwellers, followed by the Public facilities and the Recreational facilities.

From this study it can be summarized as the process of urbanization is a combination of the phenomena of isolation and integration, the status of a particular function in an area which could be measured in relation with the importance of other functions in the same area as well as the importance of the function in neighboring administrative units of the region. The urbanization is more concentrated in the coastal areas of the district with compared to the main lands and the high land regions. The trend of urbanization is more along the National Highway, State Highway and railway corridors. All the blocks in the district have almost all the infrastructural facilities to an extent but the number varies from one to the other and also the population variation in the blocks leads to a large difference in the functional index values.

References

1. Agricultural Statistics (2012-2013), Department of Economics and Statistics, Government of Kerala, Thiruvananthapuram.
2. Ansari, Abdul Sameer (2001), 'Urban renewal and development – A case study of Hyderabad' Rawat Publication – Jaipur and New Delhi.
3. District census handbook 2001, 2011 – Directorate of Census Operations, Thiruvananthapuram, Kerala.
4. District Urbanization Report (2011): Department of Town and Country Planning, Government of Kerala.
5. Guttman, L.A. (1950). The basis for scalogram analysis. In Stouffer, S.A., Guttman, L.A., & Schuman, E.A., Measurement and prediction. Volume 4 of Studies in social psychology in World War II. Princeton: Princeton University Press.
6. Krishna Chand Ranotra and D. C Kamble (2007): "A Study of Functional Classification of Towns in Maharashtra State". The Decan Geographer, Vol. 45, No. 1, June. pp. 71-82.
7. Kuruvila Yacoub Z (2012) "A Fresh look at Urbanization in Kerala: Idea for Town Panchayats", KILA – Journal of Local Governance, Vol. 1, No.1, Jan-June 2013.
8. Panchayath Level Statistics (2011): Department of Economics and Statistics, Government of Kerala, Thiruvananthapuram.
9. Sreekumar T.T (1993); "Urban Process in Kerala", Center for Development Studies Occasional paper series.
10. State Urbanization Report (2012): Department of Town and Country Planning, Government of Kerala.