RESEARCH ARTICLE

Ensemble learning for Breast Cancer prediction using Gene expression dataset

Esan Olabode Moses [1], Owolabi Olumide [2], Bisallah Hashim Ibrahim [3], Hammawa M.B [4] & Okanme Vivian Chinwe [5]

[1,2,4,5] Department of Computer Science, University of Abuja, Nigeria.
[3]Department of Computer Science, Kampala International University, Uganda
Emails: [1] astoundbme@hotmail.com, [2] olumide.owolabi@uniabuja.edu.ng, [3] hashim.bisallah@kiu.ac.ug,
[5] nnamdivivianchinwe@gmail.com

ABSTRACT

Artificial intelligence (AI) has substantially impacted several fields. The rise of deep learning (DL) together with machine learning (ML), plus the availability of enormous DNA datasets, has opened new possibilities for research targeted at applying these models for predicting breast cancer based on genetic data. Breast cancer ranks among the most prevalent and dangerous diseases affecting women, and diagnosing it early can dramatically cut the mortality rates. Due to large genetic databases, deep learning has tremendous promise for predicting breast cancer. However, predicting which genes lead to malignant cells remains tough. Identifying and extracting genes responsible for cancer is critical for accurate cancer prediction. Effective prediction can also facilitate the design and administration of targeted medications. In this work, we retrieved exons from our studies' breast cancer gene sequence. The exonic regions were extracted, used to develop two deep learning models: DNN, and bi-LSTM network. DNA series were translated using K-MER method, while class labels were represented through one-hot encoding. Model performance was evaluated using standard classification metrics. The DNN model achieved a training precision of ninety-eight point five (98.5%) with validation precision of ninety-six percent (96%), whereas the bi-LSTM model obtained a training precision of ninety-four five percent (94.5%) with validation precision of ninety-one percent (91%), indicating the effectiveness of the DNN in this context.

Keywords: Artificial Intelligence, Deep Learning, Ensemble Learning, Breast Cancer, Gene Expression

*Corresponding Author
Esan Olabode Moses; Department of Computer Science, University of Abuja, Nigeria; astoundbme@hotmail.com.

Citing this article
Esan Olabode Moses, Owolabi Olumide, Bisallah Hashim Ibrahim, Hammawa M.B, & Okanme Vivian Chinwe. Ensemble Learning for Breast Cancer Prediction Using Gene Expression Datasets. KIU J. Health Sci, 2025: 5(1);

Conflict of Interest: None is declared

1.0 Introduction

The noteworthiness of artificial intelligence (AI) across various domains cannot be overstated (Yusuf & Steve, 2021). One of the most crucial and advanced algorithms in AI, utilizing new computational methods, is machine learning-ML. The various methods in ML— supervised-ML, unsupervised-ML, and semi-supervised-ML—have yielded impactful outcomes across many areas of life and professional fields. The success of machine learning (ML) techniques largely stems from access to extensive dataset or data repositories, which facilitate the identification of complex patterns and associations(Hiba et al., 2016).

Deep learning, based on the principles of artificial neural networks, has become a formidable asset in the field of machine learning (ML).

Deep learning-DL, a branch of Artificial Intelligence, has made a substantial impact in numerous domains. DL, based on the principles of artificial neural networks, has become a formidable asset in the field of ML-the future of AI stands to be reshaped by its potential. The rapid adoption of this technology has been driven by several factors, including advances in computing power, rapid data storage, parallelization, predictive capabilities, and the automatic generation of improved high-level functions and interpretations from input data. Deep learning builds on the theoretical foundations of classical ANNs (Daniele et al., 2017). Unlike traditional ANNs, it uses multiple hidden neurons and layers (typically two or more) as architectural improvements, alongside new training paradigms (Min, Lee, & Yoon, 2017).

A leading source of mortality among women continues to be breast cancer (Ismail & Sovuthy, 2019). Timely, precise diagnosis can greatly lower mortality rates. Although Artificial Intelligence (AI) has been utilized in breast cancer diagnosis via mammograms, the introduction of the Omnic dataset has further improved AI's ability to accurately predict breast cancer.

The ability to effectively manage and treat breast cancer patients hinges on accurately diagnosing diseases and forecasting disease outcomes under varying treatment conditions. Omnic technologies offer promising predictive capabilities due to their high throughput, yet they confront numerous technical hurdles (Madhukar & Elemento, 2018). Omnic data are characterized by their high dimensionality and pose significant challenges related to potential interactions. Variability between experiments is a notable systematic issue in biological studies employing limited data samples (Li et al., 2014). Nevertheless, topical advancements in machine learning have demonstrated significant potential, and innovative computational approaches that synthesize data from diverse research hold promise for overcoming these challenges (Yusuf & Steve, 2021). The precise prediction of exons from Omnic data is regarded as having the potential to transform molecular diagnostics. A vast collection of Omnic datasets is available in public repositories, and biological experiments utilizing these datasets should make full use of this extensive resource. The National Center for Biotechnology Information (NCBI) hosts human RNA-seq samples in its thousands, encompassing broad tissues range, diseases. Since most genes do not function in isolation, integrating multiple genes may enhance predictive accuracy compared to analyzing individual genes alone.

Linear models are applicable to RNA-seq data analyses, facilitating the development of predictors by combining gene expression values with assigned weights. Specific gene features can serve as markers of biological processes that impact multiple phenotypes. Numerous studies have explored the development of complex features by leveraging biological insights derived from gene sets. (Subramanian et al., 2005), ontologies (Arbaaeen & Shah, 2021), or interaction graphs (Zarringhalam et al., 2018). Recent research has utilized unsupervised machine learning methods, such as Auto-Encoders

(AE) (Lopez et al., 2018), Principal Component Analysis (PCA) (Shen & Huang, 2006), neural network schemes, to identify relevant attributes. These approaches fall under the category of representation learning, aiming to train unsupervised models to extract complex features (Bengio, Courville, & Vincent, 2013).

One of the key objectives in bioinformatics; elucidate connection between protein structure, its function. To achieve this understanding, it's crucial to recognize that the primary amino acid sequence provides valuable structural insights. Predicting DNA sequences are crucial since proteins that share similar structures typically perform similar functions. Two common methods for determining sequence similarities are BLAST and FASTA. These methods are based on two main assumptions:

i. Familiar sequence characteristics are shared by functional elements,
ii. item placement is preserved across contrasting sequences.

Even with progress in sequence orientation techniques, computational intricacy still remains significant issue. An alignment-free method, such as the approach described in (Pinello, Bosco, & Yuan, 2014), involves feature extraction techniques, like specialized DNA representations.

Recent advancements in deep learning, particularly in parallel computing, has established it as a preferred approach for handling large datasets. Deep learning has been applied across multiple domains, such as Natural-Language-Processing (NLP), computer vision, speech recognition, voice recognition, and genomic analysis. Its impact is particularly notable in medical science, especially in genomic medicine and medical imaging**.** However, there is still a notable gap in research regarding the use of DL, for genomic prediction in breast cancer. A review conducted by Yue and Wang (2018) examines the application of advanced deep learning frameworks to genomic sequencing.

This study, examine the predictive aptness of DNN, Bi-LSTM- variation of long short-term memory (LSTM). We evaluate these models using different prediction metrics.

The structural template of this study is outlined as follows: Section two (2), Section three (3) outlines related work, review of literature; likewise, it outlines some deep learning-DL architectures—DNN, LSTM, Bi-LSTM, RNN—in genomic works. Section 4 discusses materials used for predictive model. Section five (5) describes deep learning models employed for the experiments. Section six (6) presents a detailed analysis of the experimental results and dataset. Finally, Section seven (7) presents discussion and analysis summary of models implemented in this work.

2.0 Related Work and Review of Literature

Numerous studies have investigated the use of ML procedures for forecasting breast cancer, each presenting distinct methodologies. This work examines range of ML, DL methods, comprising deep neural networks-DNN, recurrent neural networks-RNN, long short-term memory (LSTM), Bisectional LSTM, among others. It also summarizes key and up-to-date literature regarding breast cancer prediction, focusing on genomic data as well as mammogram datasets.

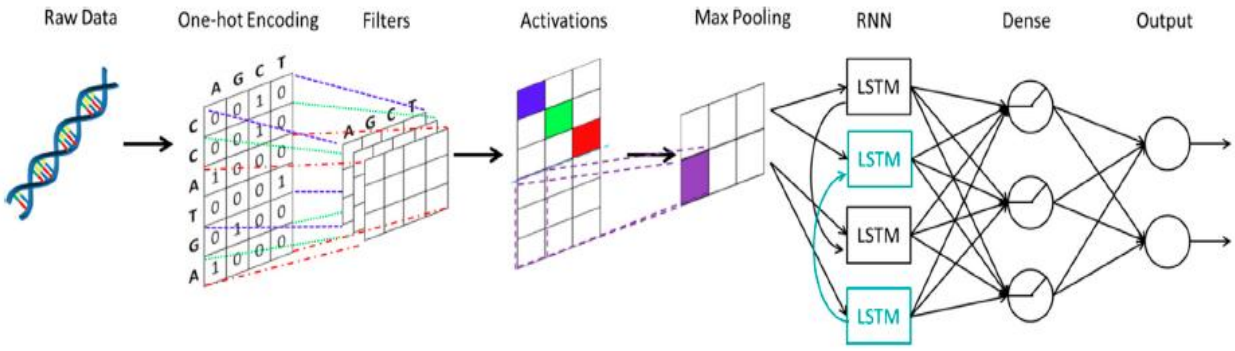

Figure 1: Showing One-hot Encoding on Genomic raw data.

2.1 Deep-NN (Neural Network)

It stimulates the functions of the human brain. Learn all the essential details: its definition, operation, applications, and training process. Neural networks are algorithmic structures modeled after the brain's neural pathways, aiming to replicate how the brain identifies patterns and transmits information across various interconnected layers. What sets a Deep Neural Network apart is its multiple-layered architecture—featuring at least two layers—enabling it to handle data in a more sophisticated manner by applying advanced mathematical techniques.

DNN draws inspiration from how the brain processes visual information across multiple stages. It mimics the way the brain's neurons interact to harness computational power. The objective is to comprehend the underlying computational principles of the brain, how it characterizes human intuition abstractly. Data enters input layer of the DNN, is processed through the activation function, then generates an output through DNN's output layer. Individual hidden layer is made of numerous neurons, with every neuron being utterly connected to the neurons in the previous layers, and these connections are governed by weights. These weights are modified during the training exercise, enabling-network to yield accurate outputs. After being trained, the network is able to generalize, perform predictions on new data. Although training DNNs can be computationally intensive, they are highly adaptable and capable of tackling complex problems, which has led to their widespread use in machine learning. Their growing popularity is also driven by the expansion of big data, advancements in processing power for parallel computations, improvements in training algorithms, and the availability of user-friendly frameworks.

2.2 Recurrent Neural Networks (RNNs)

RNNs - utilized to handle sequential data that changes over time. However, RNNs come with certain limitations. One of the key issues is that, in their basic form, each time step is assigned equal weight, leading to the input's influence on the hidden state diminishing exponentially over time. To address this drawback, an enhanced R-N-N, called “Long Short-Term Memory (LSTM)” was fostered by (Goodfellow et al., 2016).

2.3. Long Short-Term Memory-LSTM

To address the inherent limitations of traditional recurrent neural networks (RNNs), several advanced architectures have been developed—one of the most notable being the Long Short-Term Memory (LSTM) network. This design introduces a memory cell that can preserve information across extended time intervals. The cell’s operations are governed by three primary gates—input, forget, and output—which work alongside the hidden state to control the flow of data. These gates selectively manage what information should be retained, removed, or passed on, enabling the model to effectively learn long-range dependencies. This

architecture significantly mitigates issues such as the vanishing gradient problem, making LSTMs highly effective for tasks involving sequential data. (Sønderby et al., 2015).

2.4 Bi-directional LSTM

The Bidirectional Long Short-Term Memory (Bi-LSTM) model operates on the principle that the output at any time step is influenced by both preceding and succeeding elements within a sequence (Aleshinloye Abass & Arbaaeen, 2023). This structure comprises two LSTM networks running in reverse directions—one analyzing the input from start to end, and the other from end to start. By merging the contextual data from both directions, the Bi-LSTM constructs a richer and more complete understanding of the input. Unlike the conventional unidirectional LSTM, which only accounts for prior inputs, the Bi-LSTM synthesizes its hidden states and outputs by leveraging information from both temporal directions. (Desai et al., 2020)

2.5 Softmax Layer

"Recurrent Neural Networks (RNNs)" and "Long Short-Term Memory (LSTM) networks" typically require an additional output layer to handle prediction tasks. A widely used option for this purpose is the softmax layer (Bridle, 1990), which consists of k units corresponding to the number of distinct output classes. Each unit in the softmax layer is fully connected to the preceding layer and computes the probability that an input belongs to a particular class. The softmax function produces a normalized probability distribution across all classes, making it particularly effective for classification tasks and efficient during the backpropagation process.

2.6. Character Embedding

In this study, we evaluate models for genomic sequence prediction without relying on prior domain knowledge, employing various feature engineering techniques. One such technique involves k-mer analysis, which extracts all possible subsequences of length k from DNA sequences (Axelson-Fisk, 2010). The distribution of these k-mers can effectively distinguish between coding and non-coding regions of the genome. Another commonly used method is one-hot encoding, where each nucleotide is represented as a binary vector. The length of this vector corresponds to the size of the nucleotide alphabet, with a single '1' indicating the presence of a specific character and '0's elsewhere. While straightforward and widely adopted, one-hot encoding has limitations; it often lacks the expressive power needed to capture complex patterns and provides less informative features compared to representations used in other domains, such as images or audio (Krizhevsky, Sutskever, & Hinton, 2017). Moreover, unlike k-mer representations, one-hot encoding does not convey frequency-related information (Peng, Fu, & Chai, 2019).

3.0 Literature Review

Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks have been utilized in the algorithms DeepTarget (Lee et al., 2016) and DeepMirGene (Park et al., 2016) for predicting microRNA (miRNA) and target genes based on expression data. Both of these algorithms outperformed traditional models, such as TargetScan (Lewis et al., 2013). Unlike conventional approaches, DL (deep learning) methods do not require manually crafted features. The D-GEX model, for instance, employed a deep learning architecture to forecast target gene expression using landmark gene data (Singh et al., 2016). By leveraging one hundred and eleven thousand public expression profiles from the Gene Expression Omnibus, D-GEX trained a multi-layer feed-forward neural network with three hidden layers. The findings showed that this deep learning model surpassed linear regression in predicting the expression levels of approximately twenty-one thousand human genes based on around one thousand landmark genes. However, despite these

advancements, the performance of the DL model still trails behind other machine learning methods. AttentiveChrome (Lanchantin et al., 2016), an enhancement of DeepChrome developed by the same research team (Singh et al., 2016), integrates an LSTM model to enhance its capabilities. By utilizing a unified architecture, AttentiveChrome interprets the relationships among chromatin factors that influence gene expression.

Additionally, a foundational approach using a multi-layer feed-forward artificial neural network (ANN) to analyze RNA-seq gene expression data was proposed by Urda et al. (2017). This feed-forward ANN model demonstrated better performance than the Least Absolute Shrinkage and Selection Operator (LASSO) when analyzing RNA-seq gene expression profiles. Furthermore, Kuenzi et al. (2020) developed a deep learning model aimed at predicting cancer treatment responses using a pharmacogenomics dataset comprising 1,001 cancer cell lines. This deep learning strategy outperformed existing state-of-the-art machine learning methods in certain tasks. Examples of the applications of ANN, RNN, and LSTM models in genomics are summarized in Table 1.

Table 1: Application of ANN, RNN, and LSTM models in genomics

Name	Author	Dataset	
DeepNet	Lee et al. 2016	RNA- Sequence	**0.7**
DeepVariant	Kuenzi et al., 2020	Drug response with Cell-line	**Area under curve of 0.6**
Deep MirG	Park et al., 2016	MiRNA and non-miRNA	**0.89**
AttentiveChrome	Lanchantin et al., 2016	Modified Histone	**0.89**
D-GEX	Singh et al., 2016	ExLandGene	**0.8**
Deep-Target	Park et al., 2016	miRNA	**0.86**

4.0 Data Collection and Methodology

4.1 Data Gathering

The breast cancer genomic sequence data was obtained from the National Center for Biotechnology Information (NCBI), a publicly available database for nucleotide sequences. The dataset was downloaded in FASTA file format. With the vast amount of human genomic sequences now available to the public, researchers are actively exploring these datasets to identify genetic variations that could enhance our understanding of human diseases. Most variations appear as Single Nucleotide Polymorphisms (SNPs) and small insertions or deletions. Non-synonymous SNPs, in particular, are often linked to disease phenotypes, as they can result in harmful amino acid substitutions or nonsense mutations in proteins. In this study, the gene sequences sourced from NCBI were derived from tissue samples that had undergone RNA sequencing. These samples represented 95 individuals and spanned twenty-seven distinct tissue types.

For this study, we utilized nine (9) genomic datasets in our experiment, including BRCA1, BRCA2, PALB2, CDH1, CHEK2, STK11, ETS1, PTEN, and TP53. Each gene set was assigned labels of "High," "Low," or "Normal." These labels facilitated the application of deep learning predictive algorithms on the genomic data.

Different nucleotides made up a complete DNA-namely "**A**denine (**A**)", "**C**ytosine (**C**)", "**G**uanine (**G**)", and "**T**hymine (**T**)" (Anastassiou, 2001).

In this study, the exons identified from the gene sequence (BRCA1, BRCA2, PALB2, CDH1, CHEK2, STK11, ETS1, PTEN, and TP53) are obtained from the DNA sequence, as illustrated in Figure 2 below.

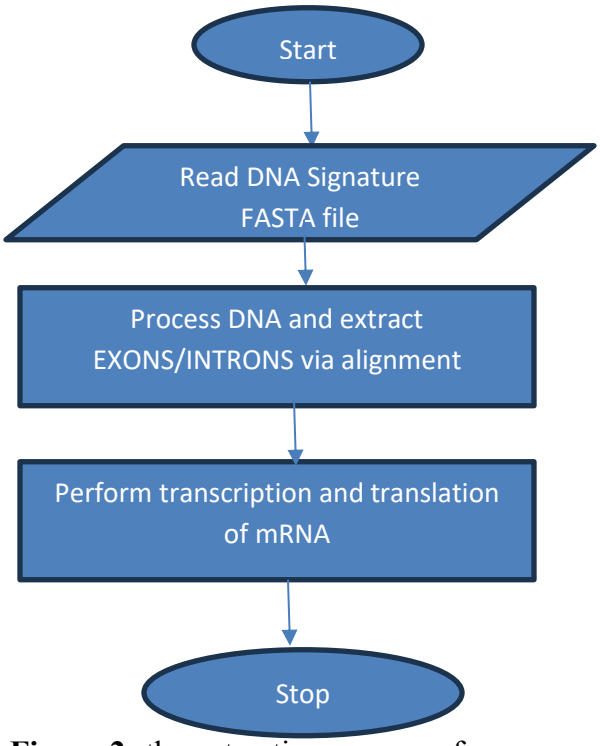

Figure 2: the extraction process of exons

The sequencing of individual genes is typically conducted at the exon level. Initially, the process involves reading the FASTA file that contains the DNA sequence specific to the gene of interest, which is sourced from the sequence database. Subsequently, the relevant genomic sequence is identified and retrieved. Once the genomic and mRNA sequences are obtained, the exon and intron regions are determined. The mRNA segment is then transcribed, with non-coding regions removed and the coding regions joined to form a continuous mRNA chain. Finally, this mRNA chain undergoes translation.

The exonic dataset is subsequently categorized according to genetic type. Figure 3 illustrates a sample dataset featuring a genomic sequence along with its corresponding class label.

	Sequence	Type	Label
0	TTTCTCAGATAACTGGGCCCCTGCGCTCAGGAGGCCTTCACCCTCT...	BRCA1-227	HIGH
1	CAAAAGCTACCCACCTTTGCCTCCTGTGCCTGCTTCTGCCCAGGGA...	BRCA1-227	HIGH
2	ATCAACTGGAATGGATGGTACAGCTGTGTGGTGCTTCTGTGGTGAA...	BRCA1-227	HIGH
3	GGTGTCCACCCAATTGTGGTTGTGCAGCCAGATGCCTGGACAGAGG...	BRCA1-227	HIGH
4	CAATTGGGCAGATGTGTGAGGCACCTGTGGTGACCCGAGAGTGGGT...	BRCA1-227	HIGH

Figure 3: Genomic sequences, types, and corresponding labels

4.2 Data Preprocessing

Data preprocessing is a crucial step in many machine learning and deep learning algorithms, especially when dealing with numerical rather than categorical data. Various methods exist for converting categorical data into numerical format. One such method is encoding, which involves transforming categorical data into numerical values. This paper explores two encoding techniques for DNA sequences: label encoding and k-mer encoding.

One-hot encoding is employed to transform DNA sequence labels into a matrix representation with dimensions $n \times l$, where n denotes the three classification labels—High, Low, and Normal. Each label is encoded as a distinct binary vector: High as [1, 0, 0, 0], Low as [0, 1, 0, 0], and Normal as [0, 0, 0, 1]. The parameter l, set to 4, defines the k-mer length, ensuring uniform encoding across all labels.

In k-mer encoding, a DNA sequence is reformatted into a structure analogous to the bag-of-words model used in natural language processing. The sequence is broken down into overlapping

substrings of length k—known as k-mers—as depicted in Figure 4. These k-mers are then sequentially joined to create a composite representation of the original sequence.

	Sequence	Type	Label	words
0	TTTCTCAGATAACTGGGCCCCTGCGCTCAGGAGGCCTTCACCCTCT...	BRCA1-227	HIGH	[tttc, ttct, tctc, ctca, tcag, caga, agat, gat...
1	CAAAAGCTACCCACCTTTGCCTCCTGTGCCTGCTTCTGCCCAGGGA...	BRCA1-227	HIGH	[caaa, aaaa, aaag, aagc, agct, gcta, ctac, tac...
2	ATCAACTGGAATGGATGGTACAGCTGTGTGGTGCTTCTGTGGTGAA...	BRCA1-227	HIGH	[atca, tcaa, caac, aact, actg, ctgg, tgga, gga...
3	GGTGTCCACCCAATTGTGGTTGTGCAGCCAGATGCCTGGACAGAGG...	BRCA1-227	HIGH	[ggtg, gtgt, tgtc, gtcc, tcca, ccac, cacc, acc...
4	CAATTGGGCAGATGTGTGAGGCACCTGTGGTGACCCGAGAGTGGGT...	BRCA1-227	HIGH	[caat, aatt, attg, ttgg, tggg, gggc, ggca, gca...

Figure 4: DNA Sequence encoding using k-mer

5.0 Models

In this research, exonic sequences of length $L1$ are segmented into overlapping k-mers of size k, producing all possible subsequences of that length. This process yields a resulting k-mer sequence of length $L_2=(L_1-k) + 1$. Two predictive models—Deep Neural Networks (DNN) and Bidirectional Long Short-Term Memory (BLSTM)—are utilized to analyze these sequences. To preserve nucleotide composition and sequence context, both label encoding and k-mer encoding strategies are applied. The generated k-mers are then used as input features, which are partitioned into training and testing datasets. The overall experimental workflow is illustrated in Figure 5.

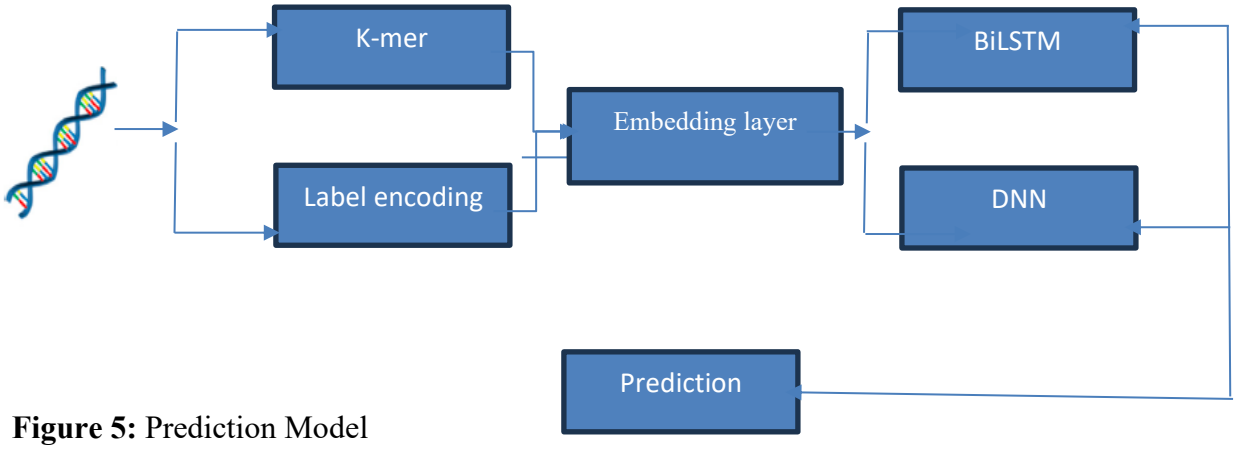

Figure 5: Prediction Model

5.1 Bi-directional Long Short-Term Memory

This study employs a bidirectional Long Short-Term Memory (Bi-LSTM) architecture to perform DNA sequence prediction. The approach combines k-mer-based feature extraction with the Bi-LSTM framework to enhance predictive performance. The model consists of two recurrent neural networks operating in opposite directions: one processes input data sequentially from start to end, while the other traverses it in reverse. This dual-directional design enables the capture of dependencies across the entire sequence. A detailed overview of the Bi-LSTM configuration is provided in Table 2. Following model initialization, training was conducted using the Adam optimization algorithm, with categorical cross-entropy as the loss function and accuracy as the evaluation metric. The structural layout of the Bi-LSTM model is depicted in Figure 6.

Table 2: Hyperparameters used for bi-LSTM model

lyperparameters	**'alues**
pochs	**0**
atch size	**2**
rchitecture function	**oftmax**
raining size	**0%**
alidation	**0%**
earning rate	**.0001**

5.2 DNN- Deep Neural Network

A four-layer sequential architecture was implemented, consisting of an input layer, an activation layer, a dropout layer, and a final output layer. The model was designed with a vocabulary size of twelve thousand eight hundred and fifty and incorporated a hidden layer comprising fifty units. A dense output layer with three nodes was employed to enable multiclass classification, utilizing the sigmoid activation function. To reduce the risk of overfitting, a dropout rate of 0.5 was applied during training. The key architectural details of the deep neural network (DNN) are outlined in Table 3. Model training leveraged the categorical cross-entropy loss function, optimized using the Adam algorithm, with accuracy used as the primary evaluation metric.

Table 3: DNN Prediction Model Hyperparameter values

lyperparameters	**'alues**
pochs	**0**
atch size	
rchitecture function	**oftmax**
raining size	**0%**
alidation	**0%**
earning rate	**.0001**

5.3 Model Training

As depicted in Figure 4, eighty percent (80%) of the input sequence is allocated to the sequential model for training. Each unit within the sequence architecture receives contextual information from the previous unit through hidden and cell state vectors. These vectors are jointly utilized to compute the model's final output. A SoftMax activation function is then employed to transform the output into a probability distribution across the target classes. To address the risk of overfitting, an early stopping criterion is applied—terminating the training process when the validation loss fails to improve over successive iterations.

5.4 Model Evaluation

In this study, deep learning approaches—namely Deep Neural Networks (DNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) architectures—were applied to predict gene sequences associated with breast cancer. These models were implemented following the methodology described in preceding sections. To encode the input features, k-mer sequences were transformed into numerical vectors using a count vectorizer, which captured the frequency distribution in a bag-of-words framework. The corresponding labels were represented using one-hot encoding. During the prediction phase, each sequence *s* was processed through both the hidden and cell states of the input layer, with the resulting features subsequently passed through a dense output layer. Visual representations of the Bi-LSTM and DNN model architectures are provided in Figures 6 and 7, respectively.

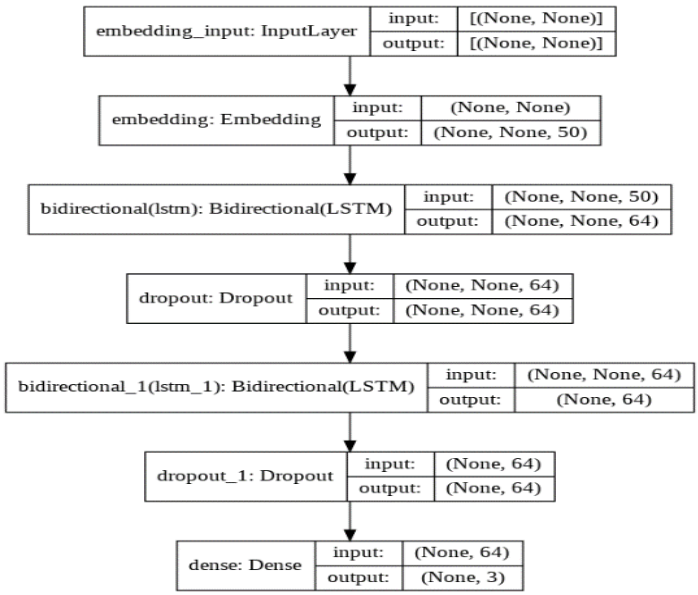

Figure 6: Bidirectional Long Short-Term

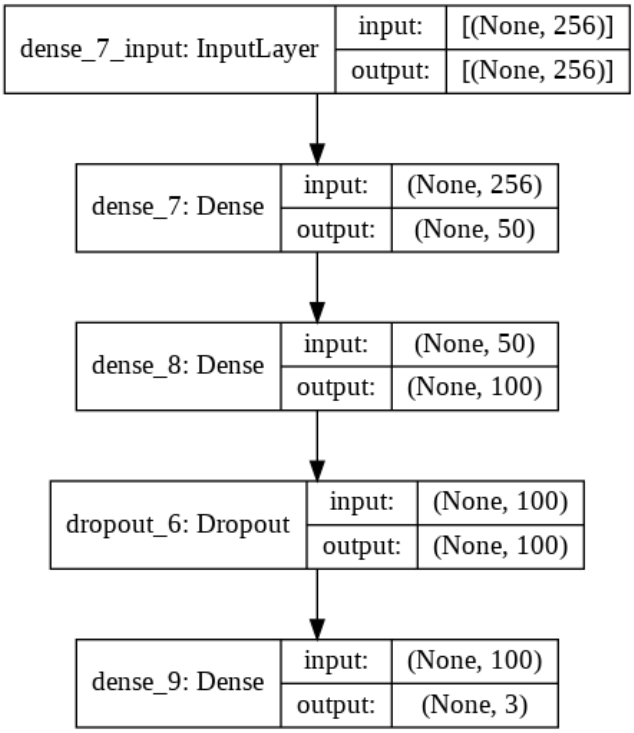

Figure 7: Deep Neural Network

6.0 Results/Outcomes

The experiments were conducted using Python 3.9 for algorithm development, and the results were produced accordingly. Additionally, Jupyter Notebook served as the text editor. The work was carried out on an HP Core i9 computer equipped with a 500GB SSD and 64GB DDRAM. A dataset, comprising thirteen thousand four hundred and thirty-eight (13,438) samples, was separated into an eighty percent (80%) training-set and twenty percent (20%) test-set, resulting in ten thousand seven hundred and fifty-one (10,751) training samples, and two thousand six hundred and eighty-six (2,686) testing samples. An experiment models were configured with a highest sequence size of 4 alongside a vocabulary size of two hundred and fifty-six. During training, the categorical cross-entropy loss function was engaged to gauge the divergence among the predicted output and target labels, guiding weight adjustments. Various hyperparameters, including the epoch quantity, layers, and embeddingmagnitudes, were tested across different models. Performance was assessed exploiting classification metrics—accuracy, precision, recall, and F1 score- derived from the confusion matrix, which provided values for True-Positive-breast-Gen (TPGe), True-Negative-breast-Gen (TNGe), False-Positive-breast-Gen (FPGe), and False-Negative-breast-Gen (FNGe). Figures 8 and 9 present the confusion matrices for the "DNN and Bi-LSTM" models, respectively, while Table seven (7) through ten (10) detail the classification metrics for both models. Figure ten (10) and eleven (11) illustrate training along with validation precision, and loss for each model.

$$\text{Accurracy} = \frac{TPGe + TNGe}{TPGe + TNGe + FPGe + FNGe}$$

$$\text{Specificity} = \frac{TNGe}{TNGe + FPGe}$$

$$\text{Sensitivity} = \frac{TNGe}{TPGe + FNGe}$$

$$\text{Precision} = \frac{TPGe}{TPGe + FPGe}$$

Table 4: Bi-LSTM training metrics

	Labels		
	High	**Low**	**Normal**
Precision	0.91	0.99	**0.95**
Recal	0.97	0.91	**0.99**
F1-Score	0.94	0.95	**0.97**
Test Accuracy			**0.945**

Table 5: Bi-LSTM test metrics

	Labels		
	High	**Low**	**Normal**
Precision	0.90	0.93	**0.87**
Recal	0.90	0.91	**0.92**
F1-Score	0.91	0.92	**0.90**
Test Accuracy			**0.91**

Table 6: DNN training metrics

	Labels		
	High	**Low**	**Normal**
Precision	0.98	0.99	**0.98**
Recal	0.99	0.98	**0.99**
F1-Score	0.98	0.99	**0.99**
Test Accuracy			**0.985**

Table 7: DNN validation metrics

	Labels		
	High	**Low**	**Normal**
Precision	0.93	0.99	**0.96**
Recal	0.99	0.96	**0.97**
F1-Score	0.98	0.92	**0.95**
Test Accuracy			**0.96**

7.0 Discussion and Analysis

The significance of each metric varies depending on the domain where it is applied. In evaluating a model's performance, “recall”, and “precision” are key metrics that data scientists in the field of data science commonly accentuate. As a major beneficiary of AI, the medical field relies on sensitivity and specificity to evaluate diagnostic tests. Though they are similar in real-world scenarios, they differ in how they are defined. “Sensitivity” evaluates the fraction of true positive sequences appropriately recognized by the model; however, when sensitivity is very high, it can sometimes result in an increased number of false positives. The harmonic mean-“recall”, “precision” is called “F1-score”. Precision measures the fraction of positive sequences (k-mers) correctly pinpointed, while specificity benchmarks the model's ability to detect negative cases. Main advantage of bi-LSTM lies in its superior performance over LSTM, as reported in (Snustad & Simmons, 2015), even though the DNN showed higher accuracy during both training and validation. However, DNNs face challenges with vanishing and exploding gradients and struggle to establish clear relationships between the parameters in tasks such as exon prediction. Exon prediction plays a pivotal role in adapted treatment, as it enables bioinformatics and medical experts to investigate gene mutations and their variations.

8.0 Conclusion

In conclusion, artificial intelligence, both “deep learning”, “machine learning”, holds great promise in advancing breast cancer prediction from genomic data. This work demonstrated the competence of two models—DNN and bi-LSTM—in predicting breast cancer based on gene sequences. The DNN outperformed the bi-LSTM, achieving higher accuracy both in training and validation. However, while these models provide significant advancements in cancer prediction, the ongoing challenge lies in identifying specific genes responsible for cancerous transformations. Overcoming this challenge is critical for the future development of targeted therapies that can further reduce breast cancer mortality.

References

1) Aleshinloye Abass, Y., & Arbaaeen, A. (2023). *Deep Learning Prediction of Exonic Sequence*. https://doi.org/10.1109/ICETAS59148.2023.10346253
2) Anastassiou, D. (2001). Genomic signal processing. *IEEE signal processing*, *8-20*.
3) Arbaaeen, A., & Shah, A. (2021). Ontology-Based Approach to Semantically Enhanced Question Answering for Closed Domain: A Review. *Information*, *12*, 200.
4) Axelson-Fisk, M. (2010). Comparative Gene Finding. *Springer*, 157-180.
5) Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, *35*, 1798-1828.
6) Bridle, J. S. (1990). Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. *Neurocomputing, Springer Book*, 227-236.
7) Daniele, R., Charence, W., Fani, D., Melissa, B., Andreu-Perez, J., Benny, L., & Guang-Zhong, Y. (2017). Deep Learning for Health Informatics. *IEEE Journal Of Biomedical And Health Informatics*, *21*(1).
8) Desai, H. P., Parameshwaran, A. P., Sunderraman, R., & Weeks, M. (2020). Comparative study using neural networks for 16S ribosomal gene classification, . *Computational Biology*, *27*, 248-258.
9) Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep Learning. *MIT Press Cambridge*, *2*.
10) Hiba, A., Hajar, M., Hassan, A., & Thomas, N. (2016). Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis. *Procedia Computer Science*, *83*, 1064-1069. https://doi.org/doi:10.1016/j.procs.2016.04.224
11) Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *ACM*, *60*, 84-90.
12) Kuenzi, B. M., Park, J., Fong, S. H., Sanchez, K. S., Lee, J., Kreisberg, J. F., Ma, J., & Ideker, T. (2020). Predicting drug response and synergy using a deep learning model of human cancer cells. *Cancer cell*, *38*, 672-684.
13) Lanchantin, J., Singh, R., Lin, Z., & Qi, Y. (2016). Deep motif: visualizing genomic sequence classifications.
14) Lee, B., Baek, J., Park, S., & Yoon, S. (2016). DeepTarget: end-to-end learning framework for microRNA target prediction using deep recurrent neural networks. *Computational Biology*, 434-442.
15) Lewis, B. P., Shih, I.-H., Jones-Rhoades, M. W., Bartel, D. P., & Burge, C. B. (2013). Prediction of mammalian microRNA targets. *Cell*, *115*, 787-798.
16) Li, S., Łabaj, P. P., Zumbo, P., Sykacek, P., Shi, W., Shi, L., Phan, J., Wu, P. Y., Wang, M., & Wang, C. (2014). Detecting and correcting systematic variation in largescale RNA sequencing data. *Nature biotechnology*, *32*, 888–895.
17) Lopez, R., Regier, J., Cole, M. B., Jordan, M. I., & Yosef, N. (2018). Deep generative modeling for single-cell transcriptomics. *Nature methods*, *15*, 1053-1058.
18) Madhukar, N. S., & Elemento, O. (2018). Bioinformatics approaches to predict drug responses from genomic sequencing. *Cancer Systems Biology*, 277–296.
19) Min, S., Lee, B., & Yoon, S. (2017). Deep learning in bioinformatics, Brief. *Bioinformatic*, *18*, 851-869.
20) Park, S., Min, S., Choi, H., & Yoon, S. (2016). deepMiRGene: deep neural network based precursor microrna prediction.
21) Peng, Q., Fu, L., & Chai, L. (2019). Predicting dna methylation states with hybrid information based deep-learning mode. *IEEE/ACM transactions on computational biology and bioinformatics*, *17*, 1721-1728.
22) Pinello, L., Bosco, G. L., & Yuan, G.-C. (2014). Applications of alignment-free methods in

epigenomics. *Briefings in Bioinformatics*, *15*, 419-430.

23) Shen, Y. J., & Huang, S. G. (2006). Improve survival prediction using principal components of gene expression data. *Genomics, proteomics & bioinformatic*, *4*, 110-119.
24) Singh, R., Lanchantin, J., Robins, G., & Qi, Y. (2016). DeepChrome: deep learning for predicting gene expression from histone modifications. *Bioinformatics*, *32*, 639-648.
25) Snustad, D. P., & Simmons, M. J. (2015). *Principles of genetics*. John Wiley & Sons.
26) Sønderby, S. K., Sønderby, C. K., Nielsen, H., & Winther, O. (2015). Convolutional LSTM networks for subcellular localization of Proteins. *International Conference on Algorithms for Computational Biology*, 68-80.
27) Subramanian, A., Tamayo, P., Mootha, V. K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, L., Golub, T. R., & Lander, E. S. (2005). Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *National Academy of Sciences*, *102*, 15545–15550.
28) Urda, D., Montes-Torres, J., Moreno, F., Franco, L., & Jerez, J. M. (2017). Deep learning to analyze RNA-seq gene expression data, in International Work-Conference on Artificial Neural Networks. *Cadiz*, 50-59.
29) Yue, T., & Wang, H. (2018). Deep learning for genomics: A concise overview. *arXiv preprint arXiv:1802.00810*.
30) Yusuf, A. A., & Steve, A. A. (2021). Deep Learning Methodologies for Genomic Data Prediction:Review. *Journal of Artificial Intelligence for Medical Science*. https://doi.org/doi.org/10.2991/jaims.d.210512.001 (Atlastis Press)
31) Zarringhalam, K., Degras, D., Brockel, C., & Ziemek, D. (2018). *Robust phenotype prediction from gene expression data using differential shrinkage of co-regulated genes*.