

Detection of Prostate Cancer using DNA Sequence analysis

CS913 – Dissertation Report

University ID: 2149607

Harshitha Suresh Vanajakshi

Supervisor: Sara Kalvala

Department of Computer Science

University of Warwick

2021-2022

Declaration

I hereby certify that this dissertation is entirely my own creation. It hasn't been submitted anywhere else for any evaluation or rewards. However, some parts of the report are from my interim report and the work is inspired by "**Aleshinloye Abass, Yusuf & Adeshina, Steve. (2021). Feature Selection with Ensemble Learning for Prostate Cancer Prediction from Gene Expression. 526. 10.22937/IJCSNS.2021.21.12.73**", the contributions have been indicated clearly and acknowledged. The research for the dissertation complies with the University of Warwick's Research Code of Practice and does not use any questionnaires or individual participant studies. To use all the software and tools in the project lawfully, I also have a student licence. All the sources of data obtained are open-source data which are freely accessible to the public and only used for research purposes. All the coding part had initially been done on the Jupyter notebook using Python and later continued on the google collab pro account due to the lack of GPU resources available on the local machine.

1. Abstract:

One of the major cancers that affects men worldwide is prostate cancer excluding some skin cancers. Among men, prostate cancer is the third most common cancer-related cause of death, making it a major global killer. Even though it is true that men who are diagnosed with prostate cancer can be cured and treated early, there are scenarios where the patients with advanced stages of cancer will have lower chances of prognosis. Advanced prostate cancer is a stage where it has been spread to most parts of the body beyond the original tumour region. Male hormones such as testosterone favours the tumour to grow and survive. Early treatment of the cancer also known as hormone therapy inhibits the growth of tumour and thus prevents from the overall spreading of the cancer. There are chances of the tumour to relapse in the future even though there are low levels of testosterone in human body and at this stage the tumour is in its advanced stage.

A key element in the successful treatment of the condition is the early detection of malignancy. It is necessary to find new and more precise prostate-specific biomarkers in order to diagnose and predict prostate cancer due to the unique limitations of the currently utilised clinical biomarkers. MicroRNAs (miRNAs), a small, non-coding species of RNA, have gained attention as potential biomarkers in cancer tissues and bodily fluids, including prostate cancer. Additionally, it has been demonstrated that miRNAs could be used as therapeutic targets in various cancer types, including prostate cancer, aiding in the improvement of diagnosis and prognosis. MiRNAs also have the potential to be clinically helpful as predictors of response to personalised cancer therapy and as predictors of prognosis.

In this study without the need for manual programming, a machine learning algorithm was built to learn from and make predictions from a given dataset retrieved from the Cosmic sanger database for cancer research (<https://cancer.sanger.ac.uk/cosmic>). A collection of RNA-Seq data from 50,000 prostate cancer patients was evaluated in this work to find transcripts associated with prostate progression. Instead of depending on tissue histology and other diagnostic techniques used in the identification of prostate cancer, machine learning with gene expression data is used to detect the presence of cancer. Supervised machine learning algorithms such as Naive Bayes (NB), Support Vector Machine (SVM), Perceptron, Logistic regression (LGR/LR), ANN (Artificial Neural Network), CNN (Convolutional Neural Network) and RNN (Recurrent Neural Network) i.e., LSTM (Long Short-Term Memory) was built using k-mer encoding for the DNA sequence and one hot encoding for the class labels and was compared against the performance metrics such as AUC-ROC (Receiver Operating characteristics), AUC-PR (Precision Recall curve) and accuracy with their mean and standard deviation scores. According to the findings, ANN performed better in terms of numerical scores, but CNN had more potential for learning and generalisation compared to other models as a classifier. The code for the dissertation can be found on github at the link given below: <https://github.com/Harshithachithu/Harshithachithu>

Keywords: Bi-LSTM, K-mer, Deep Learning, DNA Sequence, Prediction, RNN, AUC-ROC, ANN, CNN, RNN, NB

2. Acknowledgements:

I've always been curious to learn more about medical research themes and learning that University of Warwick is one of the top institutions for medical research sparked my curiosity even more. Professor Sara Kalvala coached me through some of the fundamental research methodologies during my first term and knowing that she is an expert in the subject gave me more confidence to choose her as my supervisor. I want to thank her for recommending a topic of study in one of the most significant cancer disciplines to me when I had no idea where to begin with my thesis. She has conducted meetings and offered suggestions for how to approach the problems I encountered, which has been a great help to me during my research.

I want to thank my friends to keep me motivated during my masters and helping me when needed and keeping my spirits up and being there when I needed them the most.

Last but not least, I want to express my gratitude to my parents for supporting me throughout every step of my life, from pursuing my B.E. in India to being supportive of my decision to pursue my master's in a different country thousands of miles away from them. As an international student who was autonomous and motivated to advance my profession, I had the most profound life-altering experiences.

Table of Contents:

1. Abstract:	3
2. Acknowledgements:	4
3. Introduction:	10
4. Challenges in Bioinformatics:	11
4.1 Challenges in terms of Data management and organisation:	12
4.2 Clinical challenges in Prostate Cancer:	12
5. Background:	13
5.1 Prostate Cancer Health Disparities especially in African Americans:	14
5.2 Alternative splicing:	14
6. Key Objectives of the study:	15
7. Deep Learning Models and Architecture:	16
7.1 ANN (Artificial Neural Network):	16
7.1.1 Architecture of Simple ANN:	18
7.1.2 Step-by-step working of ANN:	19
7.1.3 Backpropagation in ANN:	19
7.2 Recurrent Neural Network:	21
7.3 Long Short-Term Memory (LSTM):	22
7.4 Bidirectional LSTM:	23
7.5 CNN (Convolutional Neural Network):	24
7.5.1 Step-by-step working of General model of CNN:	24
7.5.2 Convolutional Neural Network for Text:	27
7.5.3 Convolutional Neural Network for Sequence:	28
8. Literature Review:	28
8.1 Artificial Intelligence in Prostate Cancer Digital Pathology:	29
8.1.1 Optical image analysis advancements:	29
8.1.2 Data from histopathology and demographic reports:	29
8.1.3 Performance Evaluation of AI Models and Pathologists in Prostate Biopsies (Whole Slide Image Analysis for Cancer Detection and Grading):	29
8.2 Genomics of prostate cancer using artificial intelligence:	30
8.2.1 Using mRNA and miRNA for machine learning:	30
8.2.2 Gene expression and gene activity using artificial intelligence:	30
9. Research Methodology:	36
9.1 Research Plan:	36
9.2 Data Collection and Clinical Information Retrieval:	36
9.3 Data Imputation:	37

9.4	Data Pre-processing and Character Embedding:	37
9.4.1	K-mer encoding and Label Encoding:.....	37
9.5	Model Training:.....	39
9.6	Model Validation:.....	42
9.6.1	Gradient Descent and Optimization:	42
10.	Results and Discussion:	42
10.1	Naive Bayes:	42
10.2	Support Vector Machine:.....	43
10.3	Binary Logistic Regression (LGR):.....	43
10.4	Perceptron:	44
10.5	Artificial Neural Networks:.....	44
10.6	Recurrent Neural Network:.....	46
10.7	Convolutional Neural Network:	48
11.	Conclusion:.....	51
11.1	Reflecting on the obtained results:.....	51
11.2	Numerically Predominant models:	51
11.3	Closure:	52
12.	Future Work:	52
13.	Project Management:	53
	Bibliography:	55
	Appendices:.....	65
	Graphs obtained during stratified 4-fold cross validation:	65

List of Figures:

Figure 1:Process of Alternative splicing	15
Figure 2:Typical diagram of Biological Neural Network.....	17
Figure 3:Basic Architecture of ANN	18
Figure 4: Artificial Neural Network	19
Figure 5:A recurrent neuron (left), unrolled through time (right).....	22
Figure 6:LSTM Architecture	22
Figure 7:Step-by-step walk through of LSTM.....	23
Figure 8:Typical architecture of general model of CNN.....	25
Figure 9:Illustration of convolution operation on sample image.	26
Figure 10: Pooling operation.....	26
Figure 11:One-hot encoding for sample text.....	27
Figure 12:Word embedding for sample text.....	28
Figure 13:Schematic Workflow of the study.....	36
Figure 14:Sample Data (Genomic sequences)	37
Figure 15:Showing the application of k-mer encoding to a sample sequence	38
Figure 16:DNA Sequence Prediction Model for the prognosis of Prostate Cancer	38
Figure 17:Architecture of the ANN model	39
Figure 18:Architecture of the RNN (LSTM) model	40
Figure 19:Architecture of the CNN model	41
Figure 20: Confusion Matrix for ANN.....	45
Figure 21:The plots for the validation accuracy and training accuracy alongside the training loss and validation loss before early stopping.....	46
Figure 22:The plots for the validation accuracy and training accuracy alongside the training loss and validation loss after early stopping.....	46
Figure 23:Plots showing the training and validation accuracy and training and validation loss	47
Figure 24:Confusion matrix for LSTM	48
Figure 25:The plots for the validation accuracy and training accuracy alongside the training loss and validation loss before early stopping.....	49
Figure 26:Confusion matrix for CNN	49
Figure 27:The plots for the validation accuracy and training accuracy alongside the training loss and validation loss after early stopping.....	50

List of Tables:

Table 1: Biological neural networks and artificial neural networks in relation to one another	17
Table 2:List of works showing the application of the LSTM deep learning model in genomics	34
Table 3:Showing the application of the RNN deep learning model in genomics	34
Table 4:Summary of the layers and its parameters for the ANN model	40
Table 5:Summary of all the layers with its respective parameters for the RNN (LSTM) model	40
<i>Table 6: Summary of all the layers and its parameters for the CNN model.....</i>	<i>41</i>
Table 7:Confusion Matrix for Naïve Bayes.....	43
Table 8:Summary of Performance metrics for Naive Bayes model	43
Table 9: Cost error values for linear and RBF kernel	43
Table 10:Confusion Matrix for SVM.....	43
Table 11:Summary of Performance metrics for LGR	44
Table 12:Summary of Performance metrics for Perceptron.....	44
Table 13: Hyperparameters used in ANN model	45
Table 14:Hyperparameters used in RNN model	47
Table 15:Hyperparameters for CNN model	48
Table 16:Summary of Performance metrics	50
Table 17:Project Timeline	54

Acronyms

PCa Prostate Cancer

DNA Deoxyribonucleic Acid

MiRNA Micro RNA

TNM Tumor, Node, Metastasis

PSA Prostate Specific Antigen

SVM Support Vector Machine

NB Naïve Bayes

LGR/LR Logistic Regression

DNN Deep Neural Network

KNN K-Nearest Neighbour

CNN Convolutional Neural Network

ANN Artificial Neural Network

RNN Recurrent Neural Network

LSTM Long Short-Term Memory

AS Alternative Splicing

AUC-ROC Area Under the Curve - Receiver Operator Characteristics

AUC-PR Area Under the Curve – Precision Recall Curve

PCA Principal Component Analysis

FS Feature Selection

3. Introduction:

Prostate is effectively a very large gland, and it sits between the bladder and urethra which is the pipework. It produces the fluid that washes semen and keeps the sperms healthy for successful fertilization which is the main function of prostate. When the cancer develops or begins to grow it usually begins within the edges of the prostate and can occur in more than one place, it starts to grow inside the prostate and as it advances it starts to grow through the lining or edges of the prostate.

The most typical non-cutaneous cancer in American men is prostate cancer (PCa). In 2017, almost 160000 men were diagnosed with PCa (Siegel, Miller and Jemal, 2017), and about 27000 of them passed away as a result. Given that a recent study found that the incidence of advanced PCa has increased in recent years, the disease's impact on public health is significant and is predicted to worsen (Weiner et al., 2016). Since the risk of relapse and death after therapy varies across malignancies with the same clinico-pathological parameters, such as the grade (Gleason score), stage [Tumour, Node, Metastasis (TNM)] (Edge and Compton, 2010, Amin et al., 2017) and level of prostate specific antigen (PSA) (Papsidero et al., 1980), PCa is a complicated and heterogeneous illness (D'Amico et al., 2003, Buyyounouski et al., 2012).

There are both mild and aggressive kinds of prostate cancer, which makes it a diverse illness. The most frequent malignancy among men is prostate cancer (Pernar et al., 2018). There are many different risk assessments and classifications available. To differentiate between mild and aggressive prostate tumours, each one has important restrictions (Kattan et al., 1998). Despite the fact that the majority of metastatic hormone-naïve prostate cancers (mHNPCs) show a consistent response to the initial androgen restriction therapy that targets AR signalling, progression to a castration-resistant state is unavoidable. However, with the recent release of numerous new medications leading to an improved overall survival, the therapy landscape for metastatic castration-resistant prostate cancer (mCRPC) is changing (Beer et al., 2014, de Bono et al., 2011, Scher et al., 2012, Ryan et al., 2013, Parker et al., 2013). Relapse rates are currently reported to be between 30% and 10%, with rapid disease progression, despite advancements in prostate cancer management (Punnen et al., 2013). Additionally, it has been established that variations in PSA concentrations are not a valid criterion for determining prognosis (Punnen et al., 2013). The end result of inaccurate clinical prognostic categorization is that some patients with latent cancers receive excessive treatment, while others with aggressive tumours receive inadequate treatment.

The biomedical field must devote a great deal of attention to a crucial problem called DNA sequence prediction. When compared to the more traditional regression-based models, these strategies have been found to produce more accurate findings. Predicting the gene sequence that causes malignant disorders like prostate cancer is extremely important. One of the most difficult challenges is identifying the key characteristics of a gene sequence. Extraction of the gene sequence elements that can provide insight into the type of gene mutation is crucial since it will enable successful drug creation and advance the idea of personalised medicine. The exons from the multiple prostate gene sequences used in the experiment have been retrieved in this work.

In this study, we investigate a Naive Bayes classifier, Support Vector Machine, Perceptron, Logistic Regression, ANN (Artificial Neural Network), CNN (Convolutional Neural Network) and a Recurrent Neural Network (RNN) variant's capacity for prediction referred to as the Long Short-Term Memory (LSTM) in deep learning architecture utilising a publicly available database for prostate cancer gene sequence. We employ the ANN as the preferred option due to its less training time and has better performance metrics compared to other models. On the other hand CNN which is vastly used for image classification has been explored in the form of 1D for the purpose of text analysis in conjunction with NLP processing tasks and has proved to have a better learning curve with less validation loss compared to other models with a more sophisticated computational unit that enhances performance.

4. Challenges in Bioinformatics:

The implicit objectives of bioinformatics are to read the entire genomes of living creatures, to identify each gene, to link each gene with the protein it encodes, and to analyse each protein to ascertain its structure and function using software and algorithms (Mathur, 2018). To comprehend life at the maximum level of resolution, one needs to have in-depth understanding of gene sequence, protein structure and function, and gene expression pattern (Mathur, 2018). In order to mine, analyse, compare, bioinformaticians must develop new and improved algorithms. A challenge posed by the analysis of gene expression level is for bioinformatics to create new analytical tools for a better understanding of the transition from gene to expression level and to solve the following issues (Mathur, 2018):

- To determine a protein's amino acid sequence and the protein's folding mechanism
- Finding genes and analysing genomic sequences
- Using a variety of RNA expression profiles, to infer a metabolic pathway
- Prognostication of protein structure
- Given a collection of RNA expression patterns, to determine the metabolic route

Finding out how protein structure and function are related is one of the noteworthy objectives of bioinformatics. Knowing that a large deal of valuable structural information may be gleaned from the original amino acid sequence is essential in comprehending the structure-function link. Predicting DNA sequences is crucial since sequences with comparable structures and functions also tend to have similar sequences. Sequence similarities are typically determined using the sequence alignment techniques BLAST and FASTA (Edgar, 2010).

The sequence similarities are supported by two presumptions: (a) the functional element shares common sequence properties; and (b) the order of the functional elements is maintained in different sequences (it should be noted that these assumptions are true but cannot be generalised). Computational complexity continues to be a significant obstacle despite the different developments in application approaches for sequence alignment. Alignment-free methodology is used in the study of the regulatory genome by (Pinello, Lo

Bosco and Yuan, 2013). A feature extraction phase, such as the unique representation of DNA, is part of the alignment-free method (Angelini et al., 2015).

4.1 Challenges in terms of Data management and organisation:

In the early days whenever the research was conducted by the individual groups all the data related to their work was handled by themselves since the data was less in size, but as the time progressed and the need for exploring new issues and finding solutions started there were many experimental methods that produced an enormous amount of data which almost made impossible to store and organise them using the conventional methods. Initially, these databases were just plain flat files, but for greater performance, relational databases are being employed more and more. For retrieval and content analysis, such databases now always include a web interface (Mathur, 2018). The difficulties in managing and organising data are categorised as follows (Mathur, 2018):

- How to merge information and annotation from several sequencing facilities into a searchable resource with a user-friendly interface and numerous linkages to pertinent external resources and databases under development (Mathur, 2018)
- How to combine genomic data in both graphical and text-based forms with recent and upcoming experimental findings regarding genome mapping, gene expression, protein function, protein-protein interactions, metabolic pathways and so on (Mathur, 2018)
- To ensure that everyone has complete access to the sequence and related materials regardless of location, providing web-based access and email-based access also being necessary (Mathur, 2018)

Several domains, including Natural Language Processing (NLP), Computer Vision, Speech Recognition, Voice Recognition, and Genomic Analysis have used deep learning architecture (Yue and Wang, 2018). Deep learning is now the preferred paradigm for handling big data due to developments in its architecture that allow for low-cost parallel processing (Yue and Wang, 2018). It has effectively advanced medical knowledge, especially in the areas of genetic medicine and medical imaging. Very few studies have been conducted in the field of genetic prediction of prostate cancer. In the research of (Yue and Wang, 2018), they reviewed deep learning architectures used in genomic sequencing. We will concentrate on the similar application of techniques and algorithms to explore the predictive power of the CNNs, ANNs and LSTMs, and try to make some advancements in the field of bioinformatics.

4.2 Clinical challenges in Prostate Cancer:

The initial step for all patients should be comprehensive risk stratification for males with clinically diagnosed prostate cancer. It is now more firmly established that overdiagnosis imposes an unnecessary burden on many patients, including those whose risk of mortality is relatively low (Silberstein et al., 2013). When viewed within the context of contending comorbidities in an elderly population, this becomes much apparent (Silberstein et al., 2013). Undertreatment remains far too widespread for individuals with advanced or high-risk

diseases. Adjuvant or salvage therapies are a possibility for patients who have a significant tendency of recurrence or failure following initial therapy, yet there is difference in opinion over when and how to administer these regimens most smartly (Silberstein et al., 2013). Early detection and diagnosis are incredibly challenging since the majority of prostate cancers are sluggish and show no symptoms in the preliminary phase (Sharma, Zapatero-Rodríguez and O’Kennedy, 2017). The only reliable way to diagnose prostate cancer is by performing a prostate biopsy. Medical experts commonly advise patients to have DRE (Digital Rectal Examination), TRUS (Trans-Rectal Ultrasonography), or biopsy in addition to other tests (Sharma, Zapatero-Rodríguez and O’Kennedy, 2017). Each one of these procedures, nevertheless, are often very intrusive and make patients feel uncomfortable or unpleasant (Sharma, Zapatero-Rodríguez and O’Kennedy, 2017).

However, over the past decades, the diagnosis of prostate cancer has witnessed a radical transformation due to the discovery of a highly accessible blood test for prostate-specific antigen (PSA). Men who have abnormal PSA levels frequently have a biopsy to check for prostate cancer. Following biopsy, prostate tissue is histopathologically graded using the Gleason grading system, which grades tumours from 1 to 5 (high to low differentiated) pertaining to their most predominant nature and assigns a composite index that corresponds to the aggregate of the two most prevalent patterns (Mellinger, Gleason and Bailar, 1967, Epstein, 2010, Shen and Abate-Shen, 2010). The status of the patient's primary tumours, typically originate from organ-confined to fully invasive, with or without lymph node metastases, including the presence and intensity of distant metastases, are some of the factors which are used to detect and diagnose the patient (Ohori, Wheeler and Scardino, 1994).

5. Background:

Greek anatomist Herophilus first identified the prostate gland in his corpse dissections (Bay and Bay, 2010). Antonio Ferri, a Neapolitan physician, was the first to demonstrate how malignant cells can restrict the bladder stream in 1530 (Olson, 1989a). In 1538, Vesalius created the first anatomical drawing of the prostate gland (Josef Marx and Karenberg, 2009). All these indicators reveal that individuals in those eras were familiar with the idea of prostate tumours, although George Langstaff did not report the first prostate cancer case in London until 1817 (Sharma, Zapatero-Rodríguez and O’Kennedy, 2017). The first prostate cancer case was explained by J. Adams, a surgeon at The London Hospital, in 1853 after histological analysis prompted him to its discovery (Denmeade and Isaacs, 2002). This ailment was described as "a very rare disease" by Adams in his investigation. Paradoxically, 150 years later, prostate cancer has exacerbated to become a major health concern (Denmeade and Isaacs, 2002).

5.1 Prostate Cancer Health Disparities especially in African Americans:

Over one-fifth of all newly diagnosed malignancies in males are PCa (Prostate cancer), which is the most common cancer in men in the USA (Siegel, Miller and Jemal, 2018). PCa is the second leading cause of death due to male cancer being reported every year where there are at least more than 164,000 new cases being reported (Olender and Lee, 2019). PCa also has the highest heritability of any cancer at 10% (Lynch et al., 2016). Lynch syndrome, age, race/ethnicity including the family history are one of the most recognised risk factors of PCa (Brawley, 2012, Powell, 2007). African American (AA) males are known to have significantly higher rates of PCa incidence, high-risk malignancy, and mortality than compared to non-AA men, despite increasing screening and regularly declining PCa mortality rates (Cooperberg, 2013). When compared to European American (EA) males, AA men have a mortality risk that is 2.4 times higher and are 1.7 times more likely to be diagnosed with PCa (DeSantis et al., 2016). Additionally, AA men exhibit considerably higher PSA plasma levels, more clinically advanced illness, and a three to four-fold increased risk of developing higher grade metastatic disease, suggesting that PCa appears to develop earlier in life in these individuals (Chornokur et al., 2010, Martin, Starks and Ambs, 2013, Oltean et al., 2006, Powell and Bollig-Fischer, 2013). But even after having to correct for molecular epidemiological variables, AA men continue to have significantly higher rates of morbidity and prevalence (Evans et al., 2008, Robbins, Whittemore and Thom, 2000, Tyson and Castle, 2014). Incidence, advancement, and aggressiveness of PCa may be noticeably linked to genetic ancestry, in accordance with this disease disparity.

5.2 Alternative splicing:

For growth, development, tissue homeostasis, and species variety, the procedure of alternative splicing, which involves splitting introns from pre-mRNA and uniting exons, is essential (Olender and Lee, 2019). Alternative splicing dysregulation can start and fuel disease. In both haematological and solid tumours, aberrant alternative splicing has been demonstrated to promote the "hallmarks of cancer" by hijacking and leveraging this highly complex regulated process (Olender and Lee, 2019). It is interesting to note that recent research has concentrated on the prostate cancer health disparities and the impact of alternative splicing in the disease progression (Olender and Lee, 2019). Figure 1 shows the transcriptional process in which introns are removed and exons are joined into a mature transcript.

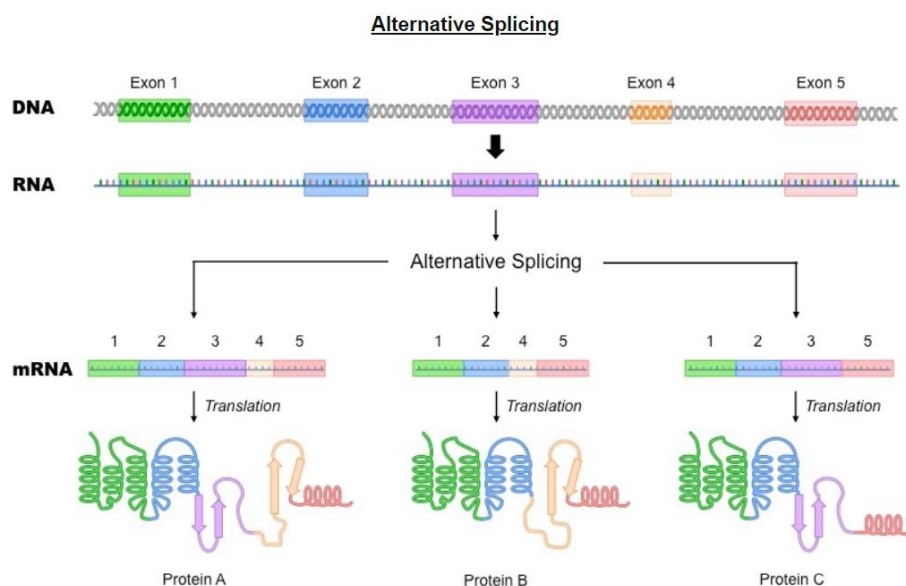


Figure 1: Process of Alternative splicing

Source: <https://ib.bioninja.com.au/higher-level/topic-7-nucleic-acids/72-transcription-and-gene/messenger-rna.html>

The main mechanism for creating protein variety is alternative splicing (AS). An increasing body of research suggests that AS contributes to the initiation and spread of malignancies. But there hasn't been a thorough examination of AS in prostate cancer. The process of producing diverse mRNA splicing bodies, which are ultimately translated into unique proteins with unique functions, from the cleavage of mRNA precursors at various places is known as alternative splicing of RNA (Zhao et al., 2020). The diversification of eukaryotes' transcriptomics and proteomics is largely a result of alternative splicing. 95 percent of human genes can be spliced, according to studies (Pan et al., 2008). One of the most prevalent malignant tumours of the male reproductive system, prostate cancer primarily affects people over the age of 65 (Salinas et al., 2014). There is mounting evidence that prostate cancer treatment targets include dysregulated cancer-specific splicing variations and splicing events as prognostic indicators (Wang et al., 2016, Li et al., 2012). Therefore, establishing a connection between splicing problems and prostate cancer is crucial for continued cancer research. Hence, it is crucial to use the genome-wide transcriptome technique to determine whether AS occurrences have any predictive value for patients with prostate cancer (Zhao et al., 2020). In our study, we will be exploring the possibilities of dysregulated variants created because of tumour cells hijacking the AS process.

6. Key Objectives of the study:

The main objective of the study is to build a number of deep learning models and optimise the traditional algorithms to get a better understanding of prostate cancer as a disease and how it would be more helpful to overcome the clinical challenges in prostate cancer. Despite technological advancements, management of prostate cancer and its early detection has become very complex. Keeping this as a motivation this study identifies the following challenges that exist in the present day:

- The Raw Data and challenges in Bioinformatics: Firstly, the Data extraction process which needed investigation of various cancer databases that are publicly available for research and choosing the appropriate sequences which uniformly and equally represents the ratio of cancerous and non-cancerous cells. It is necessary to have some knowledge on the User Interface side of the cosmic database from which the data was successfully downloaded after some professional guidance by the support team at sanger cosmic research database.
- Final Dataset and deciding if it is sufficient for the research problem: Once they were identified the actual genomic sequence from the FASTA file was mapped with the metadata file in accordance with the Gene ID helped prepare the final dataset. Accounting for various missing values by using statistical techniques such as data imputation was performed which is one of the crucial steps which aided in the better understanding of the data.
- Understanding the process of Alternative splicing: Identifying exons which are the coding region in the prostate gene sequences was a crucial step.
- Building the model and getting the best out of it: The mapped genomic sequence data is collected, and a novel framework based on better feature selection and metaheuristic learning is used. Using machine learning-based models to achieve desirable classification can be extremely effective. Each algorithm in ensemble learning serves a particular purpose and has unique qualities. Firstly, traditional algorithms have been used to identify the classification problem and an attempt has been made to exploit the features and advantages of three kinds of Neural networks including ANN, RNN (LSTM) and CNN. A special case of CNN to be used called as 1D CNN, whose applications in general as 2D or 3D would be employed in image and video analysis and its potential to solve other sequence problems in Bioinformatics.

7. Deep Learning Models and Architecture:

7.1 ANN (Artificial Neural Network):

The human brain is assimilated together into machine using deep learning, a subcategory of machine learning. It is a collection of neural network algorithms that attempts to simulate the behaviour of a human cognitive system and learn from experiences since it is inspired by how a human brain operates (Analytics Vidhya, 2021a). A computational model that replicates how nerve cells in the human brain function is termed as an artificial neuron network (or neural network). Figure 2 (left) illustrates the typical diagram of a biological neuron resembling the neural network. The biological neural network processes the information in parallel, whereas the artificial neural network processes information in series, and the former one's computational speed (measured in milliseconds) is slower than the latter one (measured in nanoseconds) (Analytics Vidhya, 2021a). These are just few of the key differences between a biological neural network and an artificial neural network (Analytics Vidhya, 2021a). Table 1 summarises the analogy between a neuron and artificial neural network.

Figure 2 (right) briefly depicts the functioning of a single neuron. Each neuron in this scenario has a set of weights and biases, and calculations are based on each of these parameters are as follows:

$$\text{combination} = \text{bias} + \text{weights} * \text{input} \Rightarrow F = w^{(1)}x^{(1)} + w^{(2)}x^{(2)} + w^{(3)}x^{(3)} \dots \dots \dots (1)$$

followed by the activation function, which yields the desired result,

$$\text{output} = \text{activation}(\text{combination}) \dots \dots \dots (2)$$

Equation 3 represents the sigmoid activation function. Other activation functions include tanh, ReLU, Leaky ReLU, and some others which can also be used at the output layer to obtain desired result.

$$\text{Sigmoid Function} = \frac{1}{1 + e^{-x}} \dots \dots \dots (3)$$

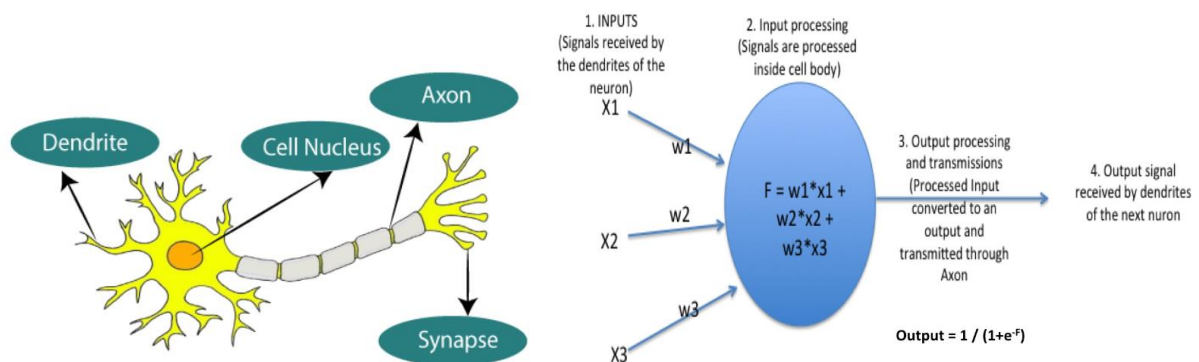


Figure 2: Typical diagram of Biological Neural Network

Source: <https://www.javatpoint.com/artificial-neural-network>

Biological Neural Network	Artificial Neural Network
Dendrites	Inputs
Cell Nucleus	Nodes
Synapse	Weights
Axon	Output

Table 1: Biological neural networks and artificial neural networks in relation to one another

Artificial neural networks (ANNs) leverage learning algorithms that facilitate them to dynamically recalibrate, in a sense, gain knowledge through learning as they are provided with preliminary data (Techopedia.com, 2019). As a result, they are an effective tool for modelling non-linear statistical data (Techopedia.com, 2019). Deep learning ANNs empower the broader scope of artificial intelligence (AI) technologies and play a pivotal role in machine learning (ML) (Techopedia.com, 2019). The primary objective of ANNs is to identify and characterize trends by demonstrating a tenuous relationship between inputs and outputs (DataFlair, 2017). These artificial neural networks have been used for a multitude of activities,

which include image recognition, speech recognition, machine translation, and medical diagnosis (DataFlair, 2017).

7.1.1 Architecture of Simple ANN:

Understanding the components of a neural network is necessary to comprehend the principle of the architecture of an artificial neural network. A substantial array of artificial neurons, also referred as units, are stacked in a series of layers to constitute what is termed as a neural network. Artificial Neural Network primarily consists of three layers as shown in Figure 3.

Input Layer: The inputs/information emanating from the outside world that are supplied to the model to learn from and make inferences from are also known as input nodes. Information from input nodes is forwarded on to the hidden layer, the subsequent layer (Analytics Vidhya, 2021b).

Hidden Layer: The collection of neurons in the hidden layer are where all calculations on the input information are performed. A neural network may comprise any number of hidden layers. One hidden layer makes up the simplest network (Analytics Vidhya, 2021b). The hidden layer can be understood of as a "distillation layer" that captures some of the key input patterns and passes them on to the next layer for observation. By separating out the unnecessary information from the inputs and recognizing only the critical information, it streamlines and improves the overall network (Jahnavi Mahanta, 2017).

Output layer: The model's conclusions or output are revealed in the output layer and are the outcome of all calculations that are performed. In the output layer, there could be single node or possibly multiple. When dealing with a binary classification problem, the number of output node is 1, however when dealing with a multi-class classification problem, the output nodes can be more than 1 (Analytics Vidhya, 2021b).

If the architecture contains multiple layers, then it is also called as Multi-layer Perceptron (MLP).

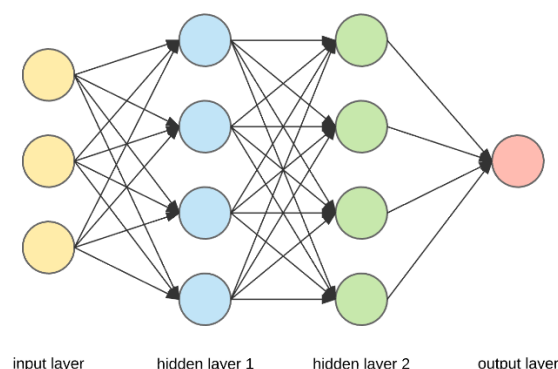


Figure 3: Basic Architecture of ANN

Source: <https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6>

7.1.2 Step-by-step working of ANN:

Note: The below section is collectively taken from (Analytics Vidhya, 2021b)

- The input values are passed in the first stage. Data is transferred to the hidden layer with some weights assigned to it. There is an option of choosing any number of hidden layers. As shown in Figure 4 inputs $x_1, x_2, x_3, \dots, x_n$ is passed.
- Hidden layer is a fully connected layer where all the inputs are connected to all of the inputs. The computation in hidden layers is segregated into two sections as listed below:
 - All inputs are multiplied by the weights assigned to them. Each variable's weight corresponds to its gradient or coefficient. It indicates how significant a specific input was. A bias component is then added after the weights are set. In order to ensure that the model fit in addition to being feasible, bias is added as a constant.
$$Z_1 = b + W_1 in_1 + W_2 in_2 + \dots + W_n in_n \dots \dots \dots (4)$$
where $W_1, W_2 \dots W_n$ are the weights assigned to the inputs $in_1, in_2 \dots in_n$
 - The linear equation Z_1 is then assigned to the activation function in the subsequent phase. Before the input is forwarded on to the next layer of neurons, it performs a nonlinear transformation known as the activation function. The activation function is crucial for facilitating nonlinearity in the model.
- Each hidden layer carries out the entire procedure that was just outlined. We proceed to the final layer, our output layer, which presents us with the final output, after passing through each hidden layer. This process is called as *forward propagation*.
- The error, which is the difference between the actual and expected output, is calculated after receiving the predictions from the output layer. If the error is significant, measures are conducted to mitigate it, and *back propagation* is also employed for this purpose.

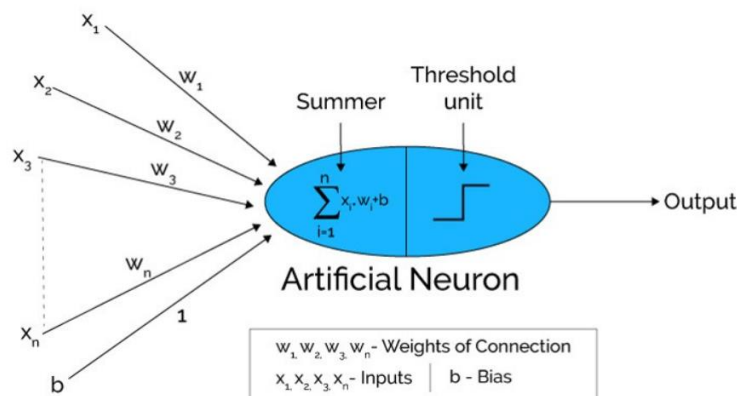


Figure 4: Artificial Neural Network

Source: <https://www.analyticsvidhya.com/blog/2021/05/beginners-guide-to-artificial-neural-network/>

7.1.3 Backpropagation in ANN:

One of the most extensively used neural network models is the back propagation (BP) neural network method, which is a multi-layer feedforward network trained using the error back propagation algorithm (Li et al., 2012a). There is no need to reveal in advance the

mathematical equation that explains these mapping relations because BP networks can learn and store a large number of input-output model mapping relations (Li et al., 2012a). Its learning rule is to employ the steepest descent approach, which controls the network's weight value and threshold value through back propagation to get the lowest error sum of squares (Li et al., 2012a).

Back-propagation is frequently used interchangeably with the whole-learning algorithm for multi-layer neural networks (Goodfellow, Bengio and Courville, 2016). The gradient is actually learned using a different technique, such as stochastic gradient descent, and back-propagation merely refers to the mechanism for computing the gradient (Goodfellow, Bengio and Courville, 2016). Furthermore, back-propagation is frequently mistaken for being unique to multi-layer neural networks when, in fact, it can theoretically compute derivatives of any function (Goodfellow, Bengio and Courville, 2016).

The Pseudo-code of the back-propagation algorithm in training ANN is given below (Guo et al., 2019):

```

Step 1: procedure – Train
Step 2: X <- Training Data set of size m x n
Step 3: y <- Labels for records in X
Step 4: w <- The weights for respective layers
Step 5: l = the number of layers in neural network, 1 ..... L
Step 6:  $D_{ij}^{(l)}$  = The error for all l, i, j
Step 7:  $t_{ij}^{(l)} = 0$ , for all l, i, j
Step 8: For i = 1 to m
Step 9:  $a^l = \text{feedforward}(x^{(i)}, w)$ 
Step 10:  $d^l = a(L) - y(i)$ 
Step 11:  $t_{ij}^{(l)} = t_{ij}^{(l)} + a_j^{(j)} t_i^{l+1}$ 
Step 12: If j ≠ 0 then
Step 13:  $D_{ij}^{(l)} = \frac{1}{m} t_{ij}^{(l)} + \lambda w_{ij}^{(l)}$ 
Step 14: else
Step 15:  $D_{ij}^{(l)} = \frac{1}{m} t_{ij}^{(l)}$ 
        where  $\frac{\partial}{\partial w_{ij}^{(l)}} J(w) = D_{ij}^{(l)}$ 

```

7.1.3.1 Mechanism of Backpropagation:

Back propagation operates in two stages:

The first phase, which is referred to as propagation, entails the four actions listed below:

- Set and initiate the neural network's weights.
- Transmit inputs across the network in order to produce output values
- Evaluate the error factor

- The error of each output and hidden neuron is obtained by propagating the feedback through the network.

The following adjustments to connection weights are applied in the second phase of back propagation:

- To calculate the gradient of the weight, the weight's output error and input are multiplied.
- The weight is reduced by a predetermined percentage determined by the weight's gradient's learning rate.

7.2 Recurrent Neural Network:

* Note: This section is collectively taken from (Goodfellow, Bengio and Courville, 2016) and (Abass and Adeshina, 2021)*

The Recurrent Neural Networks (RNN) are used to process time-evolving sequences of data. Recurrent neural networks resemble feedforward neural networks in appearance, with the exception that they also have connections pointing backward. Let's examine the most basic RNN, which consists of just one neuron that processes inputs, generates an output, and then feeds itself the output as shown in Figure 5 (left). This recurrent neuron receives its own output from the previous time step, y_{t-1} , and inputs x_t at each time step t (also known as a frame). We can plot this network along the axis of time also referred as unrolling the network through time as shown in Figure 5 (right) (Abass and Adeshina, 2021). The equations 5 and 6 below gives the full representation.

$$h_t = \theta (W_{hh}h_{t-1} + W_{xh}x_t + b_h) \dots \dots \dots (5)$$

$$y_t = \theta(W_{hy}h_t + b_y) \dots \dots \dots (6)$$

Where W_{xh} and W_{hh} are respectively the weight matrices for the input and the internal state, W_{hy} is the weight matrix for producing the output from the internal state, and the two b are bias vectors. RNN is well recognised to have various shortcomings. The preceding formulation's RNN restriction is that each time step has the same weight and that the input contribution in the hidden state is subject to exponential decay. Long Short-Term Memory (LSTM), an RNN variation, was first presented in (Goodfellow, Bengio and Courville, 2016).

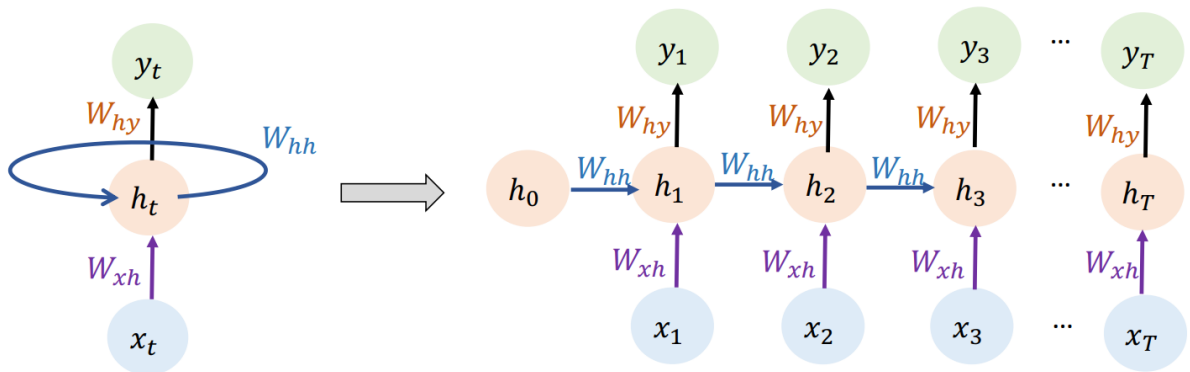


Figure 5: A recurrent neuron (left), unrolled through time (right)

7.3 Long Short-Term Memory (LSTM):

Note: This section is collectively taken from (Hochreiter and Schmidhuber, 1997), (Olah, 2015), (Feng, 2022)

To solve the flaw of the straightforward RNN, other RNN variants have been developed. A well-known RNN variation is LSTM (Hochreiter and Schmidhuber, 1997). Every LSTM unit is connected to memory, which is commonly known as a cell. An illustration of LSTM architecture is shown in Figure 6.

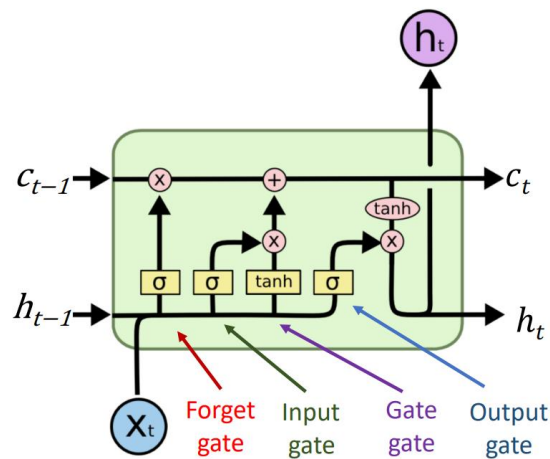


Figure 6: LSTM Architecture

LSTM has four gates namely (Feng, 2022):

- i: input gate, whether to write to cell
- f: forget gate, whether to erase the cell
- o: output gate, how much to reveal cell
- g: gate gate (?), how much to write to cell

The LSTM is thought to be effective in capturing long-term dependencies and was created expressly to address the difficulties presented by vanishing gradient concerns (Olah, 2015).

Step-by-step LSTM walkthrough as shown in Figure 7:

- Step 1: what information we are going to throw away from the cell state.

- Forget gate – outputs a number between 0 and 1
- 1: “completely keep this” - 0: “completely get rid of this.”
- Step 2: what new information we are going to store in the cell state.
 - Step 2.1: input gate – whether to write to cell. Gate gate – how much to write to cell
 - Step 2.2: Combine these two to create an update to the cell.
- Step 3: what to output based on the cell state
 - Step 3.1: output gate – decides what parts of the cell state to output.
 - Step 3.2: put the cell state through tanh (to push the values to be between -1 and 1) and multiply it by the value of output gate, so that we only output the parts we decided to

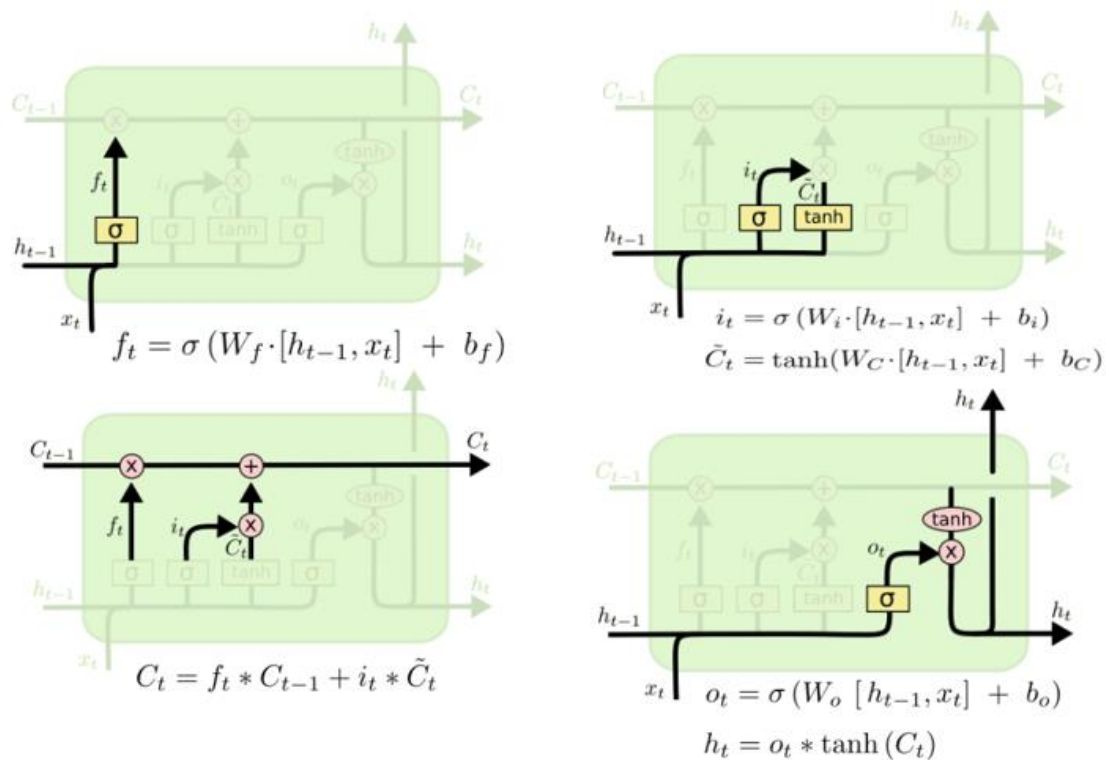


Figure 7: Step-by-step walk through of LSTM

Source: <http://colah.github.io/posts/2015-08-Understanding-LSTMs/>

7.4 Bidirectional LSTM:

* Note: This section is collectively taken from (Desai et al., 2020) and (Abass and Adeshina, 2021) *

Traditional LSTMs can be extended to create bidirectional LSTMs, which can enhance model performance for sequence classification issues. Bidirectional LSTMs train two LSTMs instead of one on the input sequence when all timesteps of the input sequence are accessible for a task. The first on a copy of the input sequence that has been reversed, and the second on the input sequence as it is. This can give the network more context and help it learn the problem more thoroughly and quickly. The concept of such a network is created by superimposing two LSTM layers on top of one another, making the output reliant on the calculation of the hidden

states from both LSTM layers as opposed to only one, as in the case of the unidirectional LSTM network (Desai et al., 2020).

7.5 CNN (Convolutional Neural Network):

As compared to fully connected neural networks Convolutional Neural Network (CNN, also referred to as ConvNet having a deep feed-forward architecture and an astoundingly superior capacity to generalise (Nebauer, 1998). Fieres, Schemmel and Meier et al., in their study CNN is referred to as the concept of hierarchical feature detectors in a biologically inspired manner. It is capable of efficiently identifying things and learning extremely abstract attributes (Zhang, 2016). The following are some of the important justifications for why CNN is preferred to other traditional models:

- The concept of using weight sharing in CNN is the main area of interest because it significantly reduces the number of parameters that need training and enhances generalisation (Arel, Rose and Karnowski, 2010).
- Less parameters allow for smoother training of CNN and thus prevents overfitting (Smirnov, Timoshenko and Andrianov, 2014).

The core concept of the convolutional neural network, a deep learning model, is the extraction of patterns from input data using convolutional layers (Nguyen et al., 2016). It was motivated by the visual mechanism of living creatures (Nguyen et al., 2016). In a model of a convolutional neural network, neurons in a convolutional layer are capable of extracting features of higher-level abstraction from features of a preceding layer (Nguyen et al., 2016).

7.5.1 Step-by-step working of General model of CNN:

CNN can be in two phases namely Training phase and inference phase. Training phase is the more involved phase where a lot of data is taken in and processed going in a feedback pattern and creating the filtered datasets which are then used in the inference phase which are mostly utilised during real-time applications.

The components of a general CNN model are (a), the convolution layer (b), pooling layer (c), activation function (d), and fully connected layer. The functionality of each layer is described below:

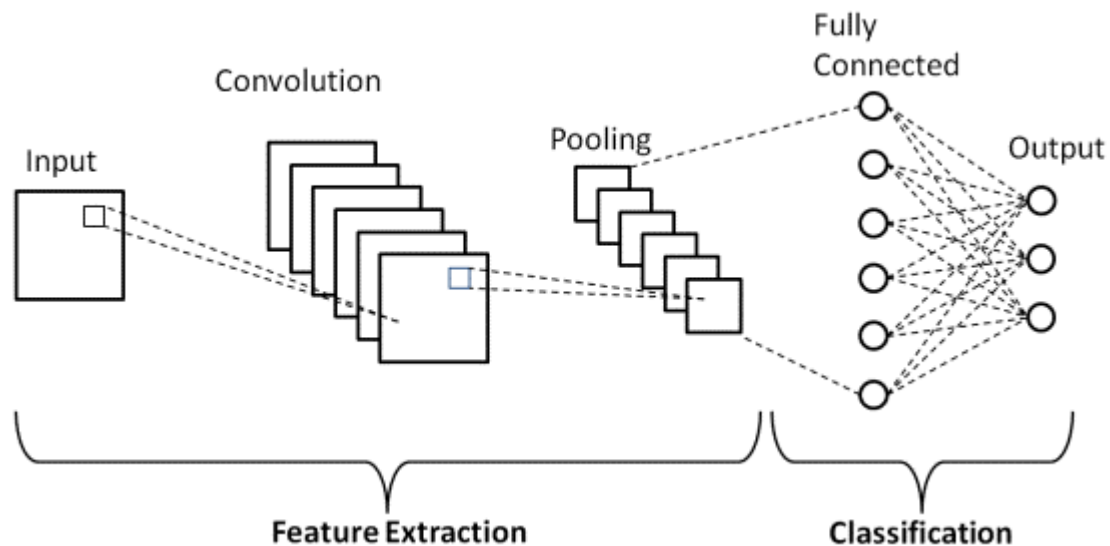


Figure 8: Typical architecture of general model of CNN.

Source: <https://www.upgrad.com/blog/basic-cnn-architecture/>

- a. **Convolutional Layer:** It is a core block of CNN. This basically consists of a set of filters with pre-defined size according to the input image size and is applied to the part of the image and then the same image will be entirely processed with number of activation filters sequentially applied. A receptive field is formed when an individual neuron in the next layer is linked to some neurons in the preceding layer (Nebauer, 1998). Using receptive field, local features from the input image are retrieved (Lecun et al., 2006) and this concept also reduces the number of trainable parameters. The weight vector is now formed due to the mechanism of receptive field of a neuron connected to a specific region in preceding layer and remains constant throughout which in turn helps recognising similar features at different coordinates in the input. The feature map is generated by sliding the weight vector, also known as the filter or kernel, over the input vector (Lee, Cheon and Kim, 2017). Convolution operation refers to the procedure of sliding the filter horizontally and vertically by taking just a patch of the given image at a time as shown in Figure 9. If we consider a coordinate (i, j) in an image, at the end of the convolution layer the output is given by the equation 7 as shown below:

$$o_{ij} = \theta((W * X)_{ij} + b) \dots \dots \dots (7)$$

where X is the input provided to the layer, W is filter or kernel, b is the bias term and * is the convolution operation and θ is the non-linearity factor introduced by the network. Since convolution operation is a linear operation, this layer is usually followed by ReLu or tanh or other functions to introduce non-linearity and hence ensures to remove overfitting probability to make it more fitting into real-life case.

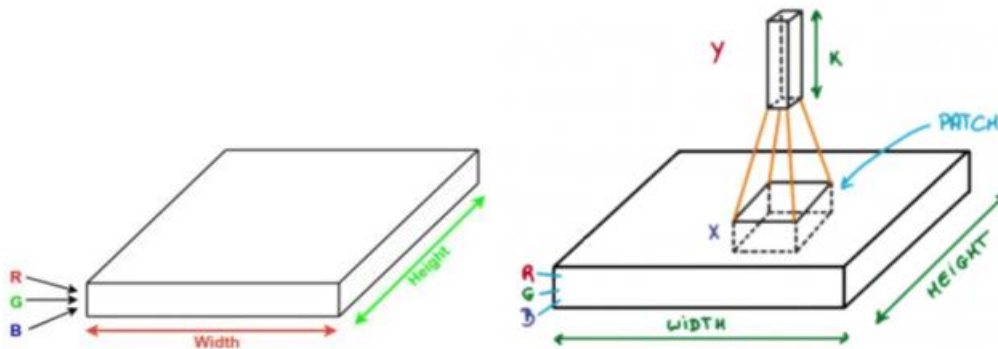


Figure 9: Illustration of convolution operation on sample image.

Source: <https://www.geeksforgeeks.org/introduction-convolution-neural-network/>

- b. Pooling layer: The feature map then goes to the pooling layer which is kind of a non-linear down sampling layer as shown in the Figure 8 where the original size of the image has been reduced. There are different aspects and non-linear functions that are applied to the down-sized image. This layer is added into the ConvNets on a regular basis, and its major job is to lower the size of the volume, which speeds up computation, saves memory, and prevents overfitting. Pooling layers are classified into two types: max pooling and average pooling. Figure 10 illustrates the max pooling operation with the stride of 2. Max pooling is most commonly used since it reduces the map-size. Pooling layer also introduces translational variance along with the reduction in the number of trainable parameters (Zhou et al., 2016). This layer is very important since CNNs are known for utilising heavy computational power and heavy memory, by introducing the concept of pooling we could reduce the computation complexity and memory complexity.

The above two layers can be repeated any number of times in order to capture the entire feature of an input image.

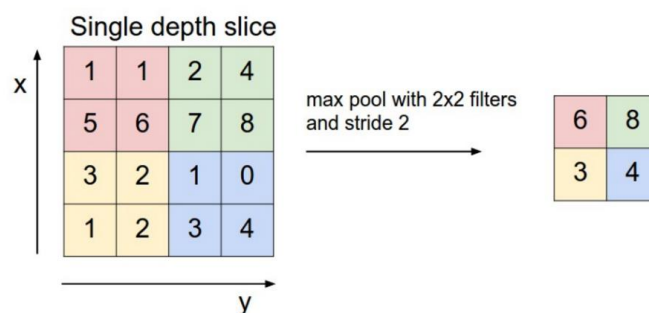


Figure 10: Pooling operation.

Source: <https://www.geeksforgeeks.org/introduction-convolution-neural-network/>

- c. Fully connected layer: As the name suggests each and every node in this layer is connected with every other node in the subsequent phase. It is a heavy data driven load and lot of parameters and co-efficients are loaded so as to support the nodes from the pooling layer data. This layer is characterised to draw collectively the most significant features from the object under consideration.

- d. Output layer: The fully connected layer is usually followed by any activation function like softmax or SVM to obtain the probability distributions and finally extract the highest probability distributions for the type of input image that are predicted as possibilities. Based on which of the possibilities has highest probability the confidence level of predicting the image will increase and is finally rendered at the output layer.

Although CNNs are popularly used for image classification task they are also extensively used for text classification task which results in good performance. Since their speciality is to extract features from the given input the same principle can be incorporated here. Based on the type of classification task say if it is an image or video analysis, a 2D CNN or 3D CNN can be used respectively. Similarly for the task of text analysis 1D CNN can be used.

7.5.2 Convolutional Neural Network for Text:

In the previous section the possibility of using CNN for image classification or video analysis was explored, one other possibility is to use 1D CNN which can be used for text classification task. The techniques that we use with the image processing like pooling can also be used for spam detection, sentiment classification etc. In many research, each word in the vocabulary is matched with a represented vector called the word vector using a lookup table. For each model, the size of the word vectors is determined, and their values are either learned as part of the convolutional neural network or adjusted via another method such as word2vec (Mikolov et al., 2013). One-hot vectors are used to represent the words which is a better approach when compared to lookup table since it has a limitation of using unigrams although bi-grams, n-grams are more efficient for the representation. The n-gram information obtained using the one-hot encoding is then integrated in the text representation by appending word vectors of some surrounding words (Nguyen et al., 2016).

Let's consider an example as show in Figure 11 having the words "I", "cat", "dog", "have", and "a" as part of a dictionary D. Each word here has a one-hot vector representation as shown below. For a sentence "I have a dog," we need to consider bi-grams taking two words at a time and append a one-hot vector to represent it in a 2-D matrix. This two-dimensional matrix is then given as input to the convolutional neural network (Nguyen et al., 2016)

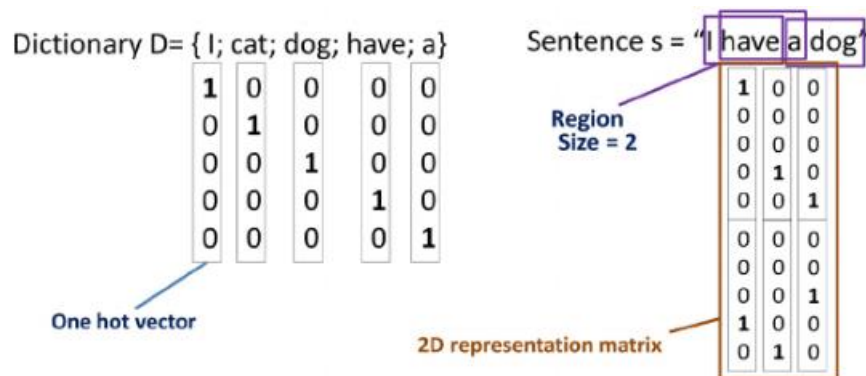


Figure 11: One-hot encoding for sample text.

Source: (Nguyen et al., 2016).

Another method that can be incorporated is word embedding where each word in the sentence is transformed systematically, deterministically into number values as shown in Figure 12. So, each word always transforms into same set of numbers. This is a manual method of generation. There is a pre-computing encoding which is much more efficient like Glove embeddings which can be used in conjunction with word2vec. Their main objective is to capture the semantic encoding of the given text and can be deemed as a transfer learning for capturing text data (Nguyen et al., 2016)

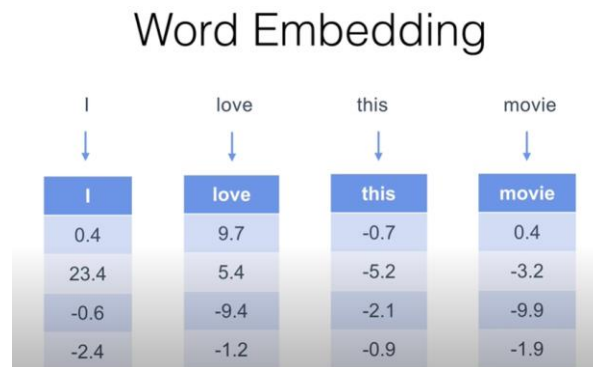


Figure 12: Word embedding for sample text

7.5.3 Convolutional Neural Network for Sequence:

DNA sequences, unlike text data, are made up of consecutive letters ('A', 'T', 'C', 'G') with no spaces in between. They are continuous sequence where there is no concept of sentence or words (Nguyen et al., 2016). As a result, we offer a method for converting DNA sequences to word sequences in order to use the same representation strategy for text data without losing position information for each nucleotide in the sequences. A method called k-mer encoding is used for this purpose (Nguyen et al., 2016).

8. Literature Review:

One of the diseases that can be fatal is prostate cancer. Over the decades there have been multiple medically critical diagnosis that has been utilised for the treatment which has proved the fact that it still has a high death rate associated with it and cannot be reduced with the current preventative treatments. Even though there exist treatments like surgery, hormone therapy, cryosurgery, Radiation therapy which are used as a monotherapy or in a multimodal approach, these come with a lot of undesirable side effects. This study outlines some of the current prostate cancer prevention, therapy and management methods that may assist to manage and avoid this seriously prevalent and life-threatening condition.

8.1 Artificial Intelligence in Prostate Cancer Digital Pathology:

8.1.1 Optical image analysis advancements:

The ML is more easily understood when viewed as a complete, including slide preservation through time, novel image handling techniques, telepathology, quality control, teaching, and collaborative research (Huisman et al., 2010). Digital histopathology slides can produce instructional sets, which are more interactive, and are simple to disseminate (virtual slide boxes). The accuracy of telepathology and image analysis using machine learning (ML), employing a variety of methods such computer assisted image analysis, is higher than that of the conventional microscope. The topics of digital imaging and digital imaging processing have been explored by Pantanowitz, 2010. Chen et al., 2019 created and assessed deep learning algorithms for the quick identification of PCa using an augmented optical light microscope that enables the real-time integration of AI. For the procedure to be optimised, more research is required.

8.1.2 Data from histopathology and demographic reports:

Lenain et al., 2019. used support vector machines (SVM), random forest (RF), and extreme gradient boosting to analyse the staging data for T (tumour), N (nodes), and M (metastasis) using histopathology data from 4470 PCa patients (XGB). They employed precision, recall, and F-Score—a metric for gauging a model's accuracy on a dataset and found the best classification of pathology results for N (F1-Score 0.98) and M. (F1-Score 0.99). A multiparametric ANN was created and verified by Roffman et al., 2018. for PCa risk categorization and prediction. The pre-histopathological condition allowed the algorithm to forecast PCa risk based on clinical and demographic traits. The ANN model's prediction of the risk of prostate cancer has a high specificity (89.4%) but a low sensitivity (23.2%).

8.1.3 Performance Evaluation of AI Models and Pathologists in Prostate Biopsies (Whole Slide Image Analysis for Cancer Detection and Grading):

Litjens et al., 2016 In order to detect PCa in the biopsy specimens, 225 glass slides of PCa biopsies were used to train a deep learning CNN, a network that can be applied to every pixel in an entire slide image. The AUC for the 90th percentile analysis of the network was 0.98, which is considered to be highly accurate for the detection of cancer on slides. Campanella, Silva and Fuchs, 2018 got results with their top model on the ResNet34 and VGG11-BN achieving 0.976 and 0.977 AUC, respectively, utilising 12,160 entire slide images, 2424 positive and 9736 negative samples, trained on the full dataset by using AlexNet and a ResNet18 network, and pretrained on several image models on ImageNet. Because PCa grading varies with urologic pathologist experience, using patient-based cross-validation and the opinions of multiple experts will improve the performance of classification methods (Melville, 1984; Nir et al., 2018). This is because patch-wise cross-validation and a single pathologist led to biases in PCa classification from histopathologic images. In the work of

Nagpal et al., 2019, a machine learning (DL) system was trained on 1557 slides, and its diagnostic performance was evaluated against a reference standard supplied by 29 pathology specialists (mean accuracy on the validation dataset was 0.61). The model achieved a higher diagnostic accuracy of 0.70 ($p = 0.002$) and is more leaning toward patient stratification risk. Lucas et al., 2019 proposed that when a CNN is trained with a proper dataset and hyperparameter tuning it will be possible for it to differentiate between benign and malignant areas with an accuracy of 92%, sensitivity of 90% and specificity of 93%.

In another study of Bulten et al., 2020 they designed a DL model to detect the Gleason grade which achieved an accuracy of 0.990 on test dataset and 0.984 on the observer dataset by analysing 5759 PCa core biopsies from 1243 patients. The technology produced identical outcomes to the reference standard and outperformed 10 out of 15 pathologists. We human beings definitely have the power of qualitatively accessing the data and on the other hand machines are good at quantitative analysis. Hence, it is needed to combine the computing power of the AI systems and the pathologists rather than utilising them on their own.

In conclusion, all available data indicates that continued development of AI systems is necessary to aid pathologists in making an accurate diagnosis. In the future, AI might offer the assistance needed for histopathological diagnosis, particularly in rural places and in healthcare systems that require the knowledge of highly qualified pathologists.

8.2 Genomics of prostate cancer using artificial intelligence:

Each individual has a different pattern of how the Prostate cancer progresses and based in the evolution of the genome associated with PCa in each individual has led to an interest in exploring the genomics of PCa. There has been enormous availability of scientific data in the last few years than before.

8.2.1 Using mRNA and miRNA for machine learning:

Lee et al., 2018 evaluated the conditions that occur after radiotherapy and predicted late toxicity using a genome-based study and pre-conditioned random forest regression and bioinformatics techniques. Only for weak stream did they produce a statistically significant prediction model ($p = 0.01$). It has been discovered that DNA methylation indicators have diagnostic potential. Bertoli, Cava and Castiglioni, 2016, using a meta-analysis method, a group of 29 miRNAs that can be used for diagnosis and a group of 7 miRNAs that may have prognostic capabilities were discovered. It is explained how miRNAs could be used as a theranostic tool in PCa. According to their findings, a prospective panel of biomarkers for the diagnosis of PC should include those 29 miRNAs mentioned in the study.

8.2.2 Gene expression and gene activity using artificial intelligence:

8.2.2.1 Statistical analysis-based model:

Using a hidden Markov model (HMM), a novel method for integrating several feature selection criteria was presented in the study of Momenzadeh, Sehhati and Rabbani, 2019. In

the suggested topology of the HMM, five feature selection ranking methods—Bhattacharyya distance, entropy, receiver operating characteristic curve, t-test, and Wilcoxon were applied. They described a method for building, learning, and inferring the HMM for gene selection that improved performance in the categorization of cancers. Three publicly accessible microarray datasets, including those for diffuse large B-cell lymphoma, leukaemia malignancy, and prostate, were examined in this experiment. From 44 gene files received relating the prognosis of breast cancer, 322 genes were chosen for their research. Results showed that, when applied to generic classifiers, the suggested HMM-based gene selection method outperformed individual feature selection criteria and Markov chain rank aggregation.

8.2.2.2 Traditional Machine Learning Algorithms:

Tavasoli et al., 2021 recommended a classification technique was employed using SVM along with hyperparameter optimization to propose a model with high-class accuracy and solving the uncertainty problems that existed in the field. The suggested method lowers the complexity of the data and the processing time by selecting ensemble features based on wrapper methods utilising five criteria. The soft weighting method uses five feature ranking techniques: Wilcoxon, two-sample T-test, Bhattacharyya distance, and receiver operating characteristic curve. A non-linear SVM with Gaussian RBF kernel was utilized along with the modified Water Cycle algorithm (mWCA), a metaheuristic technique. The end results obtained using the SVM combination with gene-mWCA was much more impactful when compared to the current models.

On three publicly available microarray cancer data sets, Glaab et al., 2012 assessed the rule-based evolutionary ML methods GAssist and BioHEL to produce a straightforward rule-based model for sample classifier. Three separate FS (Feature selection) approaches were used to classify samples in relation to other microarray standards which can compete with state-of-the-art algorithms like SVMs. With the added benefit of making interpretation easier by combining only straightforward if-then-else rules, the resultant models achieved accuracy levels above 90% in two-level external cross-validation.

K, Rajaguru and P, 2021 study's main objective is a precise classification of prostate cancer. In this work, 12,600 genes from 136 prostate samples from two classes of publicly available data were taken into account. To start with, the Kruskal-Wallis test and Principal Component Analysis (PCA) were used to identify the genes that are the most informative. Following that, these genes were categorised using Linear Discriminant Analysis (LDA), Support Vector Machine (SVM), eXtreme Gradient Boosting (XGB), and bagging ensemble KNN classifiers to categorise prostate patients as either normal or pathological. The results showed that, when compared to other types of classifiers taken into consideration in this study, the XGB classifier achieved a maximum overall accuracy of 96.46% for abnormal prostate data samples and 95.78% for normal prostate data samples.

Alkhateeb et al., 2019 developed a support vector machine (SVM) classifier to uncover biomarker genes connected to the course of prostate cancer using next-generation sequencing and the capability of machine learning. The biomarkers performed exceptionally

well at differentiating between successive stages of prostate cancer. An approach for identifying clusters of transcripts that are differentially expressed among the various Gleason stages was previously put out by Hamzeh et al., 2017. The discovered transcripts are connected with genes that are well known to have prominent roles in prostate and other types of cancer, and they can be utilised to predict the precise Gleason score for new samples. Based on gene expression patterns, Yu et al., 2004. showed that their approach is effective at assessing prostate cancer aggressiveness.

Uzma et al., 2020 introduced a gene encoder, a 2-phase FS approach that operates without supervision and is used to categorise cancer samples. They first combined three filtering techniques, including spectral-based FS (Feature Selection), PCA, and correlation approaches. The GA method, which calculated the chromosome using AE-based clustering, was then used. The classification procedure used the obtained feature subsets. To avoid relying on only one classifier, support vector machine, k-nearest neighbours, and random forest are utilised in this study. Accuracy, recall, false positive rate, precision, F-measure, and entropy are used in the comparison of performance, and it proved to be better than the current proposal.

In the study of Kumar et al., 2019, the scientists used gene expression data from The Cancer Genome Atlas and RNA sequencing to create a supervised classifier for identifying early- and late-stage prostate cancer. With the 276 most informative features culled from gene expression data, supervised learning techniques Naive Bayes, stochastic gradient descent, J48, and Random Forest, Multilayer Perceptron were used. In order to functionally classify these 276 gene characteristics, Gene Ontology and KEGG pathway enrichment analyses were also carried out.

There are a number of methods that have been proposed to classify the microarray data, including the extreme machine learning (ELM; Liu et al., 2017), ridge regression (Mohapatra, Chakravarty and Dash, 2016), relaxed least absolute shrinkage and selection operator (LASSO) and generalised multiclass SVM (Kang et al., 2019), co-expression network analysis and variational auto-encoder (Ai et al., 2020), and SVM and with LASSO (Huo et al., 2020). The categorization of gene expression has been improved in several research using effective techniques like the LASSO (Kang et al., 2019; Huo et al., 2020). The LASSO method, which is frequently a regression analysis procedure, utilised the l1-norm and Lagrange multiplier. To increase the precision and interpretability of predictions, this technique applies both feature selection and regularisation.

S A et al., 2019 - SVM, a machine learning technique, was employed in this article to explore the classification. To ascertain the precise division in separating prostate cancer cells from healthy prostate cells, SVM was compared with the Naive Bayes Classifier and Discriminant Analysis technique. The sample data used included 102 individuals with 2135 different genetic factors. According to the study's findings, the SVM method's classification has a 96% accuracy value and a 7% precision error in the tumour class. The Naive Bayes classification has an accuracy of 84% and a precision error of 23.5%. While the accuracy of the Discriminant Analysis approach is 92% with a precision error of 13.33%.

The categorization of cancer using gene expression data can be improved using a unique technique described in the research of Marchiori and Sebag, 2005. It combines support vector machines (SVM) with recursive feature selection (SVM-RFE). To create local support classifiers, the technique used pairs of support vectors from a linear SVMRFE classifier. In Naive Bayes and Optimal Bayes, this sequence served as an attributes list and an ensemble of classifiers, respectively. The leukaemia, ovarian, lymphoma, and colon cancer data were used in four publicly accessible gene expression databases.

8.2.2.3 Deep neural network-based models:

Elmarakeby et al., 2021 in their study proposed a design of P-Net, which is a biologically informed DL method which would access using method interpretability their molecular causes of treatment resistance to therapeutic targets. The model also focussed on stratifying PCa patients based on their resistance abilities to treatments of cancer at various states. By using molecular data, they showed that P-NET could predict the state of cancer more accurately than other modelling approaches.

In the study of DARENDELI and YILMAZ, 2021, they focused on using the DL approach on gene expression data to provide several views on cancer diagnoses. In this study, the normal tissue RNA-Seq data from GTEx and Cancer Genome Atlas (TCGA) as well as the RNA-Seq data from around thirty different types of cancer patients were used. The training was carried out using a CNN technique, and the input data were transformed into RGB representations. Using gene expression data, the trained algorithm was 97% accurate in predicting cancer. Their research concluded by demonstrating the enormous potential for tumour sample identification and diagnosis offered by the deep learning methodology and biological data.

Ahn et al., 2020 intended to explore how much the DL approach may pick up on in terms of identifying malignancy. They used information on gene expression from the GEO, TCGA, TARGET, and GTEx databases, comprising 13,406 data on cancer and 12,842 data on normal gene expression in 24 different tissues. The first step is to train a DNN system to distinguish between normal and malignant samples using various gene selection strategies.

DeepTarget (Lee et al., 2016) and DeepMirGene (Park et al., 2016) used the RNN and LSTM to do target prediction using expression data and micro ribonucleic acid (miRNA), respectively. The DeepTarget and DeepMirGene algorithms were used to show that miRNA predictions could be made more accurately than when using a non-DL model like TargetScan (Lewis et al., 2003). The DL model does not require any of the handcrafting components used in the earlier non-DL variants. While performing inference on gene data expression, the D-GEX developers provide a deep architecture to predict the expression of target genes from the expression data provided on the landmark gene (McDermott et al., 2020). Table 2 and Table 3 illustrates the application of LSTMs and RNNs in the field of genomics.

Another model that classified the provided gene expression into various stages or grades of prostate cancer was AttentiveChrome (Lanchantin et al., 2016), which used the Histone modification dataset. In comparison to the DeepChrome (Singh et al., 2016) model, it

performed better when measured by its AUC-ROC features. An LSTM model known as the AttentiveChrome was developed to enhance the DeepChrome's capacity to recognise the interdependencies between chromatin components that control gene expression.

Name	Publication	Dataset	Purpose	Performance	Performance Gap
DeepMirGene	(Park et al., 2016)	Positive pre-miRNA and non-miRNA	miRNA target	0.89 sensitivity	+4% f-measure
AttentiveChrome	(Lanchantin et al., 2016)	Histone modification	Classify Gene Expression	AUC = 0.81	Better than DeepChrome

Table 2:List of works showing the application of the LSTM deep learning model in genomics

Name	Publication	Dataset	Purpose	Performance	Performance Gap
DeepTarget	(Lee et al., 2016)	miRNA – mRNA pairing	prediction	0.96 Accuracy	+25% f-measure
D-GEX	(McDermott et al., 2020)	Expression of Landmark genes	Gene Expression Inference	An overall error of 0.3204 ± 0.0879	Outperform linear regression and KNN in most of the target genes

Table 3:Showing the application of the RNN deep learning model in genomics

Using a total of 111,000 public expression profiles from Gene expression Omnibus, D-GEX (McDermott et al., 2020) trained a multi-layer feed-forward deep neural network with three hidden layers. The results showed that the deep learning model worked more accurately than linear regression when estimating the expression of human genes (approximately 21000) based on the landmark genes (about 1000). The deep learning model's poor performance means that there is still potential for improvement, even though it is more accurate than other machine learning models already in use.

Azizi et al., 2018 suggested to explicitly represent the temporal information in temporal enhanced ultrasound using deep recurrent neural networks (RNN) (TeUS). They found that long short-term memory (LSTM) networks attain the highest accuracy in differentiating cancer from benign tissue in the prostate by examining numerous RNN models. Their analysis contained information from 157 participants' 255 prostate biopsy cores. Their findings suggested that, in comparison to previously published efforts, temporal modelling of TeUS using RNN can greatly increase the accuracy of cancer detection.

Long short-term memory (LSTM) and Residual Net (ResNet 101), which are not dependent on manually created characteristics, were used in the research of Iqbal et al., 2021. With the help of classifiers that do not use deep learning, such as support vector machines (SVM), Gaussian kernels, k-nearest neighbor-Cosine (KNN Cosine), kernel naive Bayes, decision trees (DT), and RUSBoost trees, the results were compared with manually created features like texture, morphology, and grey level co-occurrence matrix (GLCM). Sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), accuracy (AC), Mathews Correlation Coefficient (MCC), and area under the curve were used to evaluate the performance (AUC). The most impressive results were produced with GLCM features and KNN-Cosine, employing non-deep learning techniques.

Wu et al., 2022 illustrated that when trained using a sizable but imperfect real-world dataset of tumour marker (TM) values, they investigated the usefulness of many machine learning methods to forecast cancer risk. Data on TM screening were gathered from two different medical centres' large asymptomatic cohort (n = 163,174). 785 people in the cohort received a cancer diagnosis in the future. Long short-term memory (LSTM), a machine learning technique, has proven to be superior to other models at handling erratic medical data. They also came to the conclusion that even with widely variable test intervals, the introduction of time-series TM data can enhance the prediction performance of LSTM models.

A deep learning architecture is presented in Feng et al., 2019 to identify prostate cancer in a series of CEUS raw dataset containing 78277 negative sample and 9073 positive sample. The suggested technique, which captures the dynamic information of the perfusion process recorded in many consecutive frames for prostate cancer detection, evenly pulls features from the spatial and temporal dimensions by executing three-dimensional convolution operations. It extracts both feature temporal and the spatial by performing 3D CNN operation which overall gives 90% of an average accuracy rate.

Kwak and Hewitt, 2017 - During the tissue microarrays research programme, four tissue microarrays (TMAs) were taken from the National Institutes of Health. TMAs contains 162 tissue sample (72 benign & 89 cancer in TMA A), similarly in TMA B it contains 149 sample (76 benign & 73 cancer) and TMA C contain 157 (73 benign & 86 cancer). There were two steps to the strategy. The first phase involved performing tissue segmentation to isolate lumens in digitalized pictures of prostate tissue specimens. A multiview boosting method was used to collaboratively integrate the image data from several scales and identify lumens. Intensity- and texture-based image features were computed at five different scales. Convolutional neural networks (CNN) were used in the second step to automatically extract high-level visual properties of lumens and to anticipate malignancies.

For estimating Next-Generation Sequencing (NGS) depth from DNA probe sequences, Zhang et al., 2021 introduced a deep learning model (DLM). A bidirectional recurrent neural network that accepts both DNA nucleotide identities and the estimated likelihood of the nucleotide being unpaired as inputs is part of the DLM. Three separate NGS panels—a 39,145-plex panel for human SNPs, a 2000-plex panel for human lncRNA, and a 7373-plex panel focusing on non-human sequences for DNA information storage—were each subjected to DLM. They also

proposed that the measured single-plex kinetic rate constants for DNA strand displacement and hybridization could be accurately predicted using the same model.

9. Research Methodology:

9.1 Research Plan:

In this study, a new and improved version of already existing work has been proposed and developed for the detection and classification of PCa (Prostate Cancer) as shown in Figure 13. We will be looking at combining the functionalities of three powerful deep learning techniques to make the model predictions. The proposed plan incorporates different processes namely Data collection, Data cleaning and preprocessing, Adamoptimiser based hyper-parameter tuning, DNN (Deep neural network) based classification for reducing the computational complexity and improving the classification accuracy. The models that are chosen for the task are a simple ANN (Artificial neural network), a 1D CNN (Convolutional Neural Network) and an LSTM (Long Short-term Memory) i.e. a special case of RNN.

(Note: All of the pre-processing steps will be done on Google collab pro using Python.)

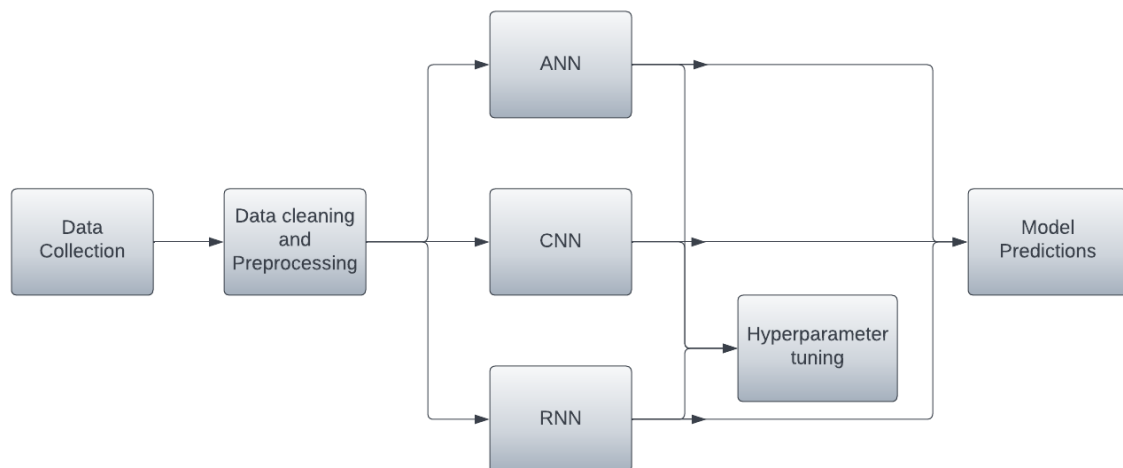


Figure 13:Schematic Workflow of the study

9.2 Data Collection and Clinical Information Retrieval:

The gene sequences under study were retrieved from the Cosmic Sanger website which is a publicly available database with nucleotide sequences. There was a total of 55173 sequences that were mapped with the obtained clinical dataset. The sequences were downloaded as a FASTA file. Clinical tumour stage information was collected from clinical data files, and gene expression. Only tumour conditions' gene expression data were used for the remaining analysis. A matrix was built based on the common Gene ID column using the patient's clinical tumour stage and gene expression data, and this matrix was then used for feature selection.

9.3 Data Imputation:

As a part of data imputation either the existing Python programming language libraries were used to apply replacement by deep learning imputation, or a direct replacement of the missing data with mean or median or mode based on the central dependencies was done, depending on how well the various imputation techniques performed. The Gene Accession number was framed into a readable format which improved the performance further. Attributes like age, tumour origin, fathmm score were replaced with the predicted values after the application of deep learning models belonging to the respective classes. The values were imputed for each of the 30 repeats during algorithm training, and for comparison of the overall model performance, the average of the AUC was taken.

9.4 Data Pre-processing and Character Embedding:

Individual gene sequencing is often done at the exon level. According to the gene of interest, which is found in the sequence database as explained in Section 5.2, the FASTA file containing the DNA sequence is first read (Abass and Adeshina, 2021). The relevant genetic sequence is then located and obtained. After obtaining the genomic and mRNA sequences, the exon/intron is determined. The part of mRNA is transcribed, the portion of mRNA that does not code for proteins is deleted, and the portion of mRNA that codes for protein is joined to form a long chain of mRNA (Abass and Adeshina, 2021). The lengthy mRNA chain is then translated. A sample data after translation corresponding to the gene of interest is shown in the Figure 14.

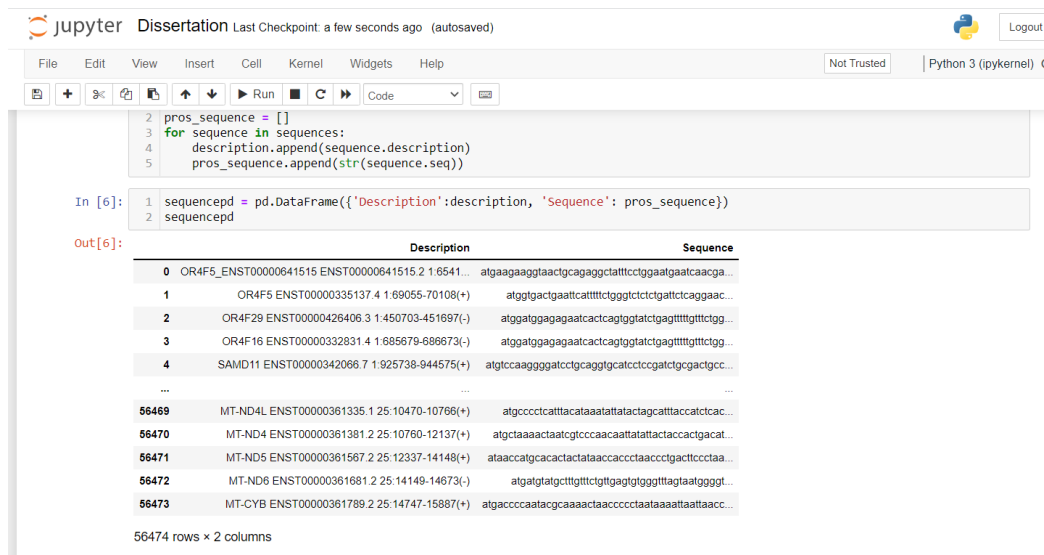


Figure 14: Sample Data (Genomic sequences)

9.4.1 K-mer encoding and Label Encoding:

In bacteria, viruses, and living cells, DNA plays the most significant chemical role. The four distinct nucleotides that make up DNA are adenine (A), cytosine (C), guanine (G), and thymine (T). Only one particular region of the DNA molecule known as the gene carries the coding

information for protein production. In this study, we test feature-engineered algorithms that can predict genomic sequences without any prior knowledge. The usage of the k-mer distribution is one strategy for doing this. Unique sub sequences consisting of A,C,G,T of a specific length k from longer DNA sequences are known as k-mers. In this work, the idea of applying k-mer is to split the given DNA sequence of length 'L' into k-mer of size 6 so as to obtain a total of L-k+1 sequences. A sample illustration of how the k-mer counting is performed is shown in the Figure 15 below.

Additionally, label encoding has been done for the prediction of cancer column. "1" representing the malignancy and "0" benign condition. Using label encoding and k-mer methods, the nucleotide information of the DNA sequence is to be preserved. The training and test sets of the k-mer are used to represent the characteristics. Figure 16 depicts the experimentation's methodology.

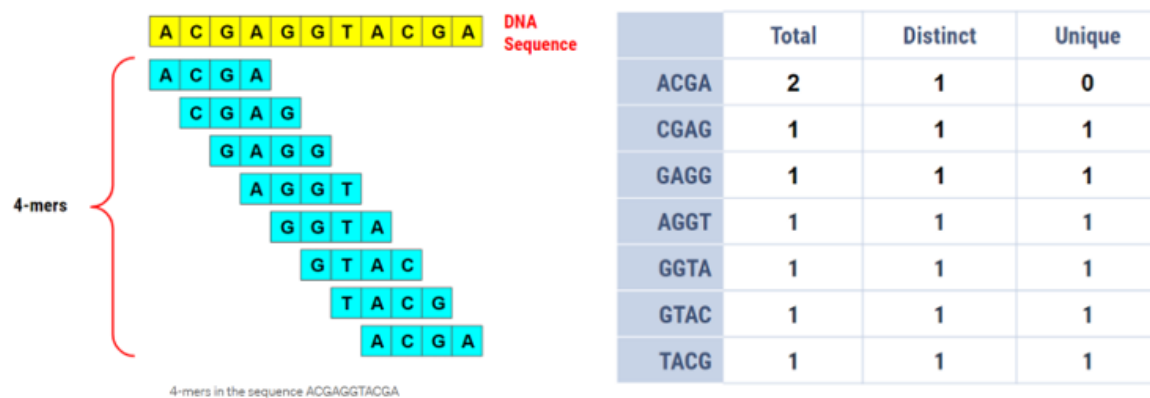


Figure 15: Showing the application of k-mer encoding to a sample sequence

Source: <https://medium.com/swlh/bioinformatics-1-k-mer-counting-8c1283a07e29>

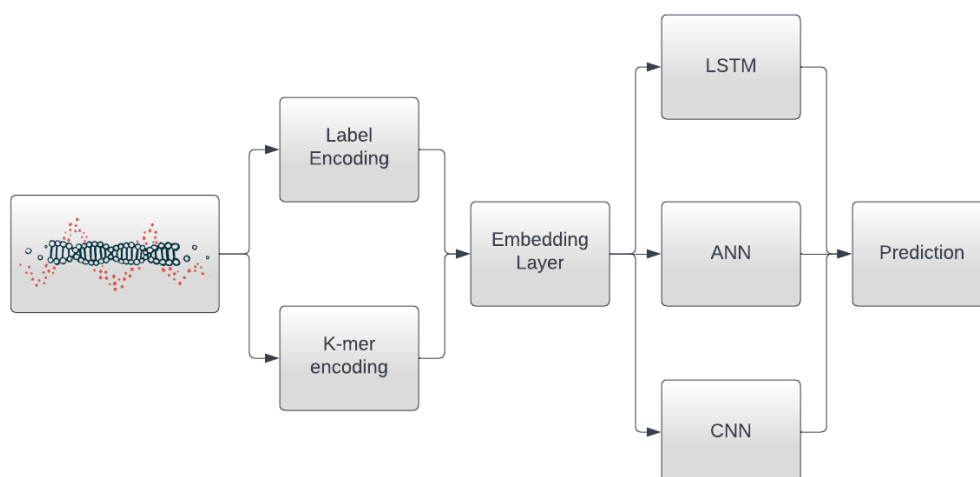


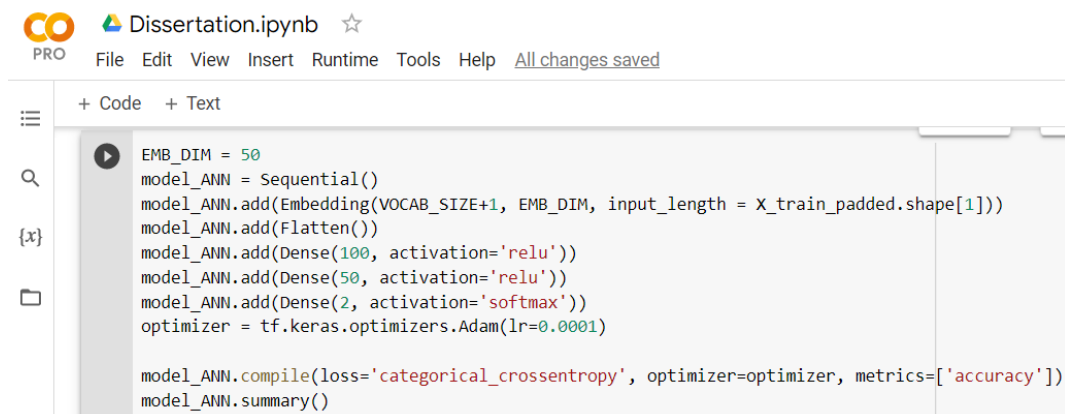
Figure 16: DNA Sequence Prediction Model for the prognosis of Prostate Cancer

9.5 Model Training:

The following stages were used to categorise genetic alterations in prostate cells:

- Preparing the data and dividing it into training, validation, testing data with a ratio of 70:10:20.
- Classification using the Naive Bayes Classifier:
 - Determining the criteria and probabilities.
 - Examining the Naive Bayes.
 - Establishing the classification's outcome:
- Classification using the Support Vector Machine technique.
 - Choosing kernel functions for modelling linear and gaussian RBF.
 - Producing the best kernel function from the smallest error value.
 - Using testing data and the best kernel, form the classification findings.
 - Determining the performance of categorization accuracy.
- Classification using ANN:
 - Developing a sequential model with five layers: input, activation, embedding, dense and output.
 - Embedding the network layer with the vocabulary size of 4127 unique tokens
 - Two Dense layers with relu as activation function and an output layer with softmax as an activation function.

The Figure 17 and Table 4 below shows the entire architecture of ANN which was a part of building the model.



```

Dissertation.ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

EMB_DIM = 50
model_ANN = Sequential()
model_ANN.add(Embedding(VOCAB_SIZE+1, EMB_DIM, input_length = X_train_padded.shape[1]))
model_ANN.add(Flatten())
model_ANN.add(Dense(100, activation='relu'))
model_ANN.add(Dense(50, activation='relu'))
model_ANN.add(Dense(2, activation='softmax'))
optimizer = tf.keras.optimizers.Adam(lr=0.0001)

model_ANN.compile(loss='categorical_crossentropy', optimizer=optimizer, metrics=['accuracy'])
model_ANN.summary()

```

Figure 17:Architecture of the ANN model

Layer (type)	Output Shape	Param #
Embedding (Embedding)	(None, 12606, 50)	206400
Flatten (Flatten)	(None, 630300)	0
dense (Dense)	(None, 100)	63030100
dense (Dense)	(None, 50)	5050
dense (Dense)	(None, 2)	102

=====

Total params: 63,241,652
 Trainable params: 63,241,652
 Non-trainable params: 0

Table 4: Summary of the layers and its parameters for the ANN model

e. Classification using RNN:

- Developing a sequential model with four layers: input, activation, dropout, and output.
- Embedding the network layer with the vocabulary size of 4127 unique tokens
- An output layer with two outputs and a softmax activation function was inserted, as well as two hidden layers of 50 neurons.
- Dropout layer rate was set at a value of 0.45 to prevent the overfitting.
- A dense layer was also added.
- Adam optimizer was used

The Figure 18 and Table 5 below shows the entire architecture of ANN which was a part of building the model.

```

[ ] LSTM_UNITS = 50

model_RNN = Sequential()
model_RNN.add(Embedding(VOCAB_SIZE+1, EMB_DIM, input_length = X_train_padded.shape[1]))
model_RNN.add(LSTM(units = LSTM_UNITS, return_sequences = True))
model_RNN.add(LSTM(units = LSTM_UNITS))
model_RNN.add(Dense(CLASS_NUM, activation = 'softmax'))

model_RNN.compile(loss = 'categorical_crossentropy',
                  optimizer = 'adam', metrics = ['accuracy'])
model_RNN.summary()

```

Figure 18: Architecture of the RNN (LSTM) model

Layer (type)	Output Shape	Param #
Embedding (Embedding)	(None, 12606, 50)	206400
lstm (LSTM)	(None, 12606, 50)	20200
lstm (LSTM)	(None, 50)	20200
dense (Dense)	(None, 2)	102

=====

Total params: 246,902
Trainable params: 246,902
Non-trainable params: 0

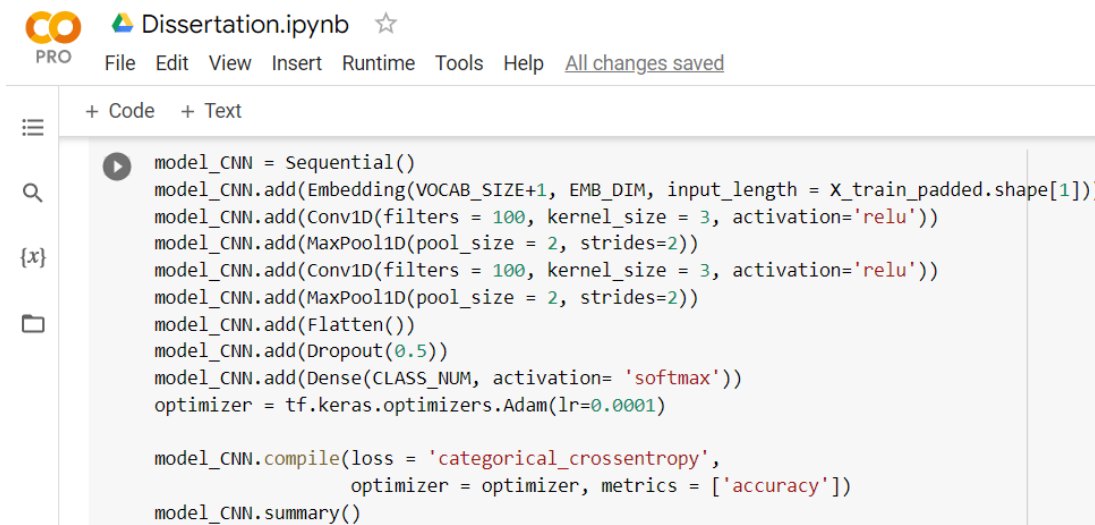
Table 5: Summary of all the layers with its respective parameters for the RNN (LSTM) model

f. Classification using CNN:

- Developing a sequential model with four layers: input, activation, embedding, convolution, maxpooling, dropout, and output.
- Embedding the network layer with the vocabulary size of 4127 unique tokens

- A convolution layer with 100 filters having kernel size as 3x3 followed by ReLu activation function
- A maxpooling layer with pooling size 2x2 and having stride of 2
- An output layer with two outputs and a softmax activation function was inserted, as well as two hidden layers with 50 neurons.
- Dropout layer rate was set at a value of 0.5 to prevent the overfitting.
- A dense layer was also added.
- Adam optimizer was used.

The Figure 19 and Table 6 below shows the entire architecture of CNN which was a part of building the model.



```

model_CNN = Sequential()
model_CNN.add(Embedding(VOCAB_SIZE+1, EMB_DIM, input_length = X_train_padded.shape[1]))
model_CNN.add(Conv1D(filters = 100, kernel_size = 3, activation='relu'))
model_CNN.add(MaxPool1D(pool_size = 2, strides=2))
model_CNN.add(Conv1D(filters = 100, kernel_size = 3, activation='relu'))
model_CNN.add(MaxPool1D(pool_size = 2, strides=2))
model_CNN.add(Flatten())
model_CNN.add(Dropout(0.5))
model_CNN.add(Dense(CLASS_NUM, activation= 'softmax'))
optimizer = tf.keras.optimizers.Adam(lr=0.0001)

model_CNN.compile(loss = 'categorical_crossentropy',
                  optimizer = optimizer, metrics = ['accuracy'])
model_CNN.summary()

```

Figure 19:Architecture of the CNN model

Layer (type)	Output Shape	Param #
Embedding (Embedding)	(None, 12606, 50)	206400
Conv1d (Conv1D)	(None, 12604, 100)	15100
Max_pooling1d (MaxPooling1D)	(None, 6302, 100)	0
Conv1d (Conv1D)	(None, 6300, 100)	30100
Max_pooling1d (MaxPooling1D)	(None, 3150, 100)	0
Flatten (Flatten)	(None, 315000)	0
Dropout (Dropout)	(None, 315000)	0
Dense (Dense)	(None, 2)	630002

```

=====
Total params: 881,602
Trainable params: 881,602
Non-trainable params: 0

```

Table 6: Summary of all the layers and its parameters for the CNN model

9.6 Model Validation:

To simulate the prediction of the prostate cancer gene sequence, Naive Bayes, Support Vector Machine, Logistic regression, Perceptron, deep learning models such as RNN (LSTM – Long Short-Term Memory), ANN (Artificial Neural Network), CNN (Convolutional Neural Network) were employed. As stated in the previous sections, all the models were utilised. To create a vector space based on the word frequency, a count vectorizer was employed to transform the bag of words (K-mer sequences). Label encoding was applied to the labels. To ensure least overfitting as possible stratified cross validation has been incorporated with four folds and the performance metrics such as precision, recall, f1-score, accuracy, AUC-PR and AUC-ROC are analysed.

9.6.1 Gradient Descent and Optimization:

As a big part of Model Validation and Evaluation, optimization is one of the trivial tasks which is necessarily performed to get the best performance for the model. With any simple basic model like linear regression, we optimise the slope and the intercept, when we use logistic regression, we optimise a squiggle and similarly with other models. The objective of gradient descent (GD) is to optimise these parameters and many more by means of calculating the loss function and minimising it. In the case of minimization, optimization techniques have one of the following goals:

- Find the objective function's global minimum. If the objective function is convex, then any local minimum is a global minimum.
- Find the target function's lowest possible value within its vicinity. That is often the situation when the objective function is not convex, which is the case in most deep learning issues.

10. Results and Discussion:

The dataset considered for the study had the data of 55173 patients out of which 34514 patients had a “Neutral” status of cancer whereas 20659 patients had a “Pathogenic” status. The nucleotide sequence data was cleaned and pre-processed as described in the earlier sections using various packages available in Scikit learn on Google Collab Pro using Python to bring it into a matrix form which is acceptable by the machine learning model. Then the dataset was divided into training data, validation data and testing data with the ratio of 70:10:20. Furthermore, classification was done using Perceptron, Logistic Regression, Naive Bayes, SVM, RNN (i.e. LSTM), ANN and CNN. The results obtained for each of the models have been discussed in detail in the next section.

10.1 Naive Bayes:

The foundation of Naive Bayes is the simplification that, if output values are provided, attribute values are conditionally independent of one another. In other words, given the

output value, the likelihood of witnessing the attributes simultaneously is a sum of the aforementioned probabilities. Table 7 and Table 8 summarises the confusion matrix obtained for the testing data. The cross validation with 5-fold was performed to avoid overfitting of the testing data.

Multinomial Naïve Bayes		Predicted	
		0	1
Actual	0	5016	1915
	1	1037	3067

Table 7: Confusion Matrix for Naïve Bayes

	Standard Deviation Accuracy	Standard Deviation AUC-ROC	Standard Deviation AUC-PR
Naive Bayes	0.015	0.019	0.028
	Mean Accuracy	Mean AUC-ROC	Mean AUC-PR
	0.7434	0.8127	0.7187

Table 8: Summary of Performance metrics for Naive Bayes model

10.2 Support Vector Machine:

The SVM approach was used for the analysis, and the kernel functions used include linear, polynomial, radial, and sigmoid but decision was made to just consider linear and RBF kernels. Choosing the kernel functions to be utilised for modelling is the first stage, and training data is used to do this. Each output for linear SVM and Guassian RBF kernel is its minimal error value as shown in Table 9. Clearly, it can be seen, that the RBF kernel would be the best choice. Table 10 shows the confusion matrix obtained for the testing data.

	Kernel	
Error value	Linear	Guassian RBF
	0.1736	0.3301

Table 9: Cost error values for linear and RBF kernel

Support Vector machine		Predicted	
		0	1
Actual	0	5240	1691
	1	844	3260

Table 10: Confusion Matrix for SVM

10.3 Binary Logistic Regression (LGR):

When the dependent variable is nominal, the LGR is a series of statistical procedures used to evaluate hypotheses or causal links. Logistic regression (LR) is mostly used for classification problems where a binary value either 0 or 1 is obtained for malignant or benign cancer conditions. LR is usually used to predict the probability of the dependent variable by analysing the relationship between the independent variables and the dependent variable. In our case,

the dependent variable is to predict malignant or benign condition for the given DNA sequence as the independent variable. It uses a logistic function to determine the probability which needs to be decided based on a threshold value set. In usual cases this value is 0.5 and any value above 0.5 is rounded off to 1 and anything below 0.5 is 0. This value can be manipulated to manage the number of false positives and false negatives. In our case we set it to 0.5. Table 11 shown below summarises the performance metrics obtained for the model.

	Standard Deviation Accuracy	Standard Deviation Mean AUC-ROC	Standard Deviation Mean AUC-PR
Perceptron	0.003	0.004	0.003
Logistic Regression	0.004	0.003	0.004

Table 11:Summary of Performance metrics for LGR

10.4 Perceptron:

The perceptron is a technique for supervised learning of binary classifiers in machine learning. It is a particular kind of linear classifier, or an algorithm for classifying data that bases its predictions on a linear predictor function that combines a set of weights with the feature vector. Several components involved in a Perceptron are the inputs, weights, bias, weighted summation, step/activation function and the output. The input DNA sequence is our feature vector with randomly set weight values which gets updated after each training error epoch. In order to translate its decision boundary, a classifier uses the bias term. Moving each point, a fixed distance in a predetermined direction and by reducing the error the bias term aids in a quicker and higher-quality model training process. In this study we obtained the results as shown in the Table 12 below.

	Mean Accuracy	Mean AUC-ROC	Mean AUC-PR
Perceptron	0.8561	0.8984	0.8565
Logistic Regression	0.8405	0.919	0.8819

Table 12:Summary of Performance metrics for Perceptron

10.5 Artificial Neural Networks:

Perceptron are single-layer neural networks, whereas neural networks are multi-layer perceptron. The performance between a single neuron as perceptron has been compared with the simple ANN constructed using four layers. ANN has three layers: Embedding layer where the input has been padded and fed, two hidden layers with 100 and 50 neurons each are employed in the architecture along with the output layer. Tensorflow, a Google library, was used to build deep learning applications, and Python was the chosen programming language. The back propagation algorithm used gradient descent, the activation function for hidden layers was the ReLu function, the loss function was the cross-entropy error function, and the output function was the softmax function. In order to prevent over-fitting, the learning rate for each step was set to 0.0001. These values had been adopted in an initial assessment of ANNs.

The network is handled in two stages, i.e. training and testing, in order to create an ANN. On the basis of input data, the network is trained for an output prediction at the training level. The hyperparameters used in the ANN is as shown in the Table 13 below. This artificial neural network (ANN) is designed to identify the presence of prostate cancer (PCa) based on DNA sequence information. Values for the ANN neuron and bias weight were determined using the training data set. By adjusting the number of neurons and the epoch number, training was repeated until the error level was as low as possible. The test data set was then subjected to the trained algorithm.

Hyperparameters	Values
Epochs	100
Batch size	32
Activation Function	Softmax, Relu
Validation size	10%
Training size	70%
Testing size	20%
Learning rate	0.0001

Table 13: Hyperparameters used in ANN model

The confusion matrix was plotted to get the true prediction rate for the malignant and benign conditions as shown in the Figure 20 below.

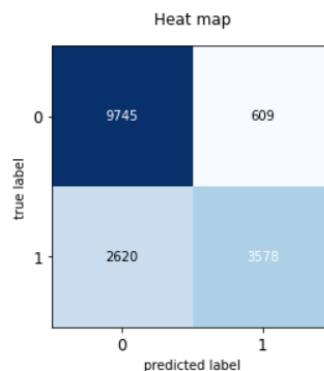


Figure 20: Confusion Matrix for ANN

The plots for the validation accuracy and training accuracy alongside the training loss and validation loss is as shown in the Figure 21. It can be seen that the validation accuracy as well as the training accuracy seems to be saturated after running for certain number of epochs and stops learning once the threshold for it is crossed. The early stopping was implemented to maintain the condition of optimisation to exactly determine the stage at which the model stops learning. The plots after adding early stopping as a parameter in the *callback* attribute is as shown in the Figure 22 below. The hyperparameter tuning was performed on various parameters before deciding on the final accuracy, precision, recall and F1-score.

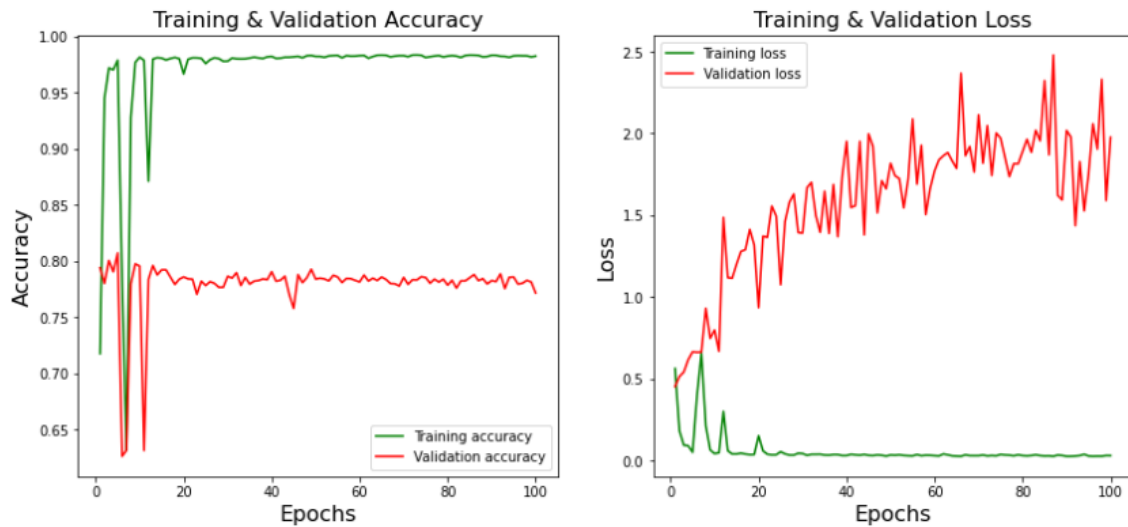


Figure 21: The plots for the validation accuracy and training accuracy alongside the training loss and validation loss before early stopping

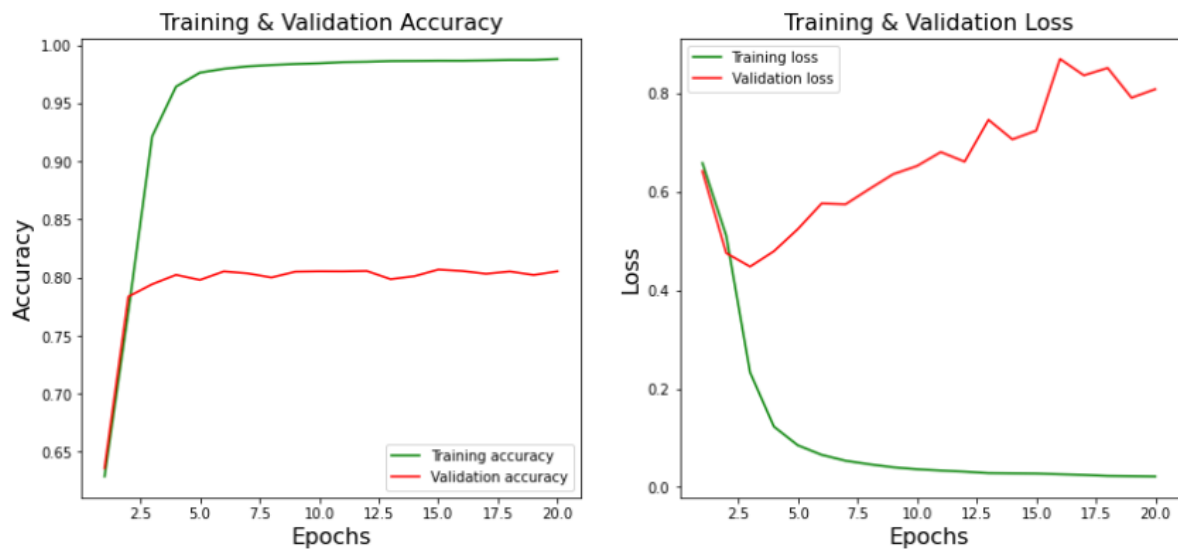


Figure 22: The plots for the validation accuracy and training accuracy alongside the training loss and validation loss after early stopping

10.6 Recurrent Neural Network:

A sequential model was built with four layers as described in the previous section with four layers. A list of the features of the RNN deep learning model is provided in Table 14. An Adam optimiser, accuracy metrics, and categorical cross entropy loss were added to the model after it had been created to improve its ability to make predictions. The loss function's objective is to compute the error between the target label and the actual output, for which the weight is trained and updated. By changing the hyper-parameters, such as the number of layers, epochs, and embedding size, the implemented RNN model was put to the test.

Hyperparameters	Values
Epochs	10
Batch size	32
Activation Function	Softmax, Relu
Validation size	10%
Training size	70%
Testing size	20%
Learning rate	0.001
Drop-out rate	0.45

Table 14:Hyperparameters used in RNN model

The model's performance for training data and the validation data was plotted for the whole of the 10 epochs as shown in the Figure 23 to analyse its performance to see if early-stopping is required. With the implementation of early-stopping and further hyperparameter tuning the overfitting of the data was avoided. The confusion matrix was plotted to get the true prediction rate for the malignant and benign conditions as shown in the Figure 24 below.

LSTM was built and run-on Google collab pro for only 10 epochs due to its long training periods which seemed unnecessary when compared to other models that were built. The amount of computation power and GPU resources were massive and for that reason the model was just considered to run for 10 epochs and analyse its performance. From the graphs it is clear, that the training accuracy is much more than the validation accuracy leading to overfitting. The same trend has been followed in the training and validation loss plot where the validation loss increases and never comes down concluding that the model is overfitting.

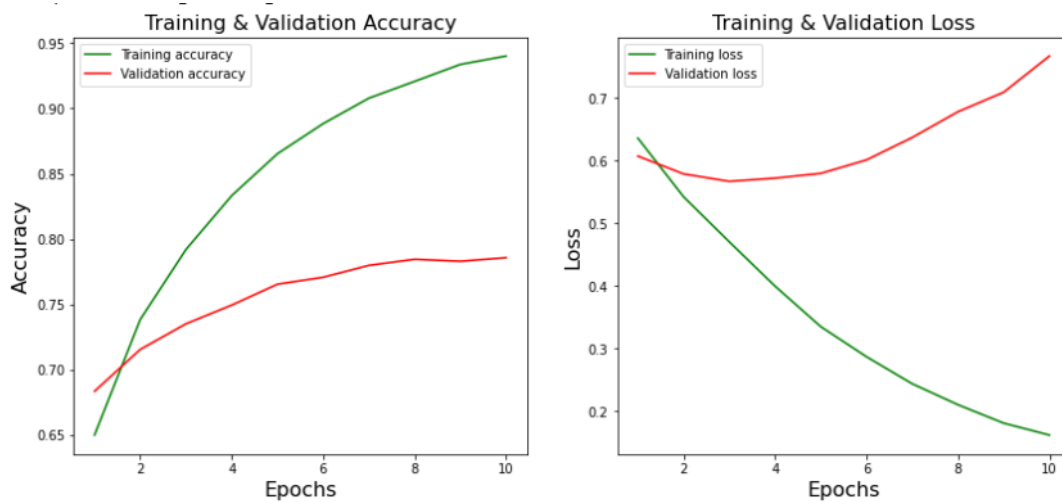


Figure 23:Plots showing the training and validation accuracy and training and validation loss

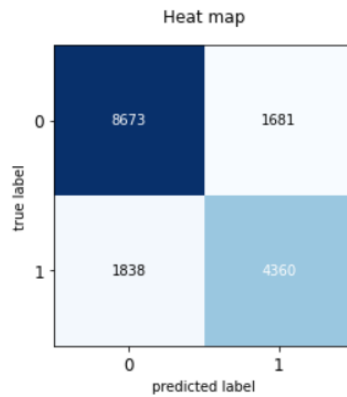


Figure 24:Confusion matrix for LSTM

10.7 Convolutional Neural Network:

To extract features from the input dataset, the CNN architecture employs a succession of convolutional layers. After each convolutional layer, the max pooling layer is added, and the extracted feature dimensions are lowered. The size of the kernel in the convolutional layer is important for function extraction. The number of filters and kernel size are the model's hyperparameters. The summary of the entire architecture has already been discussed in the model training stage. The Table 15 below shows the hyperparameters used during the model training.

Hyperparameters	Values
Epochs	100
Batch size	32
Activation Function	Softmax, Relu
Validation size	10%
Training size	70%
Testing size	20%
Learning rate	0.0001
Drop-out rate	0.5

Table 15:Hyperparameters for CNN model

The embedded layer, which has a 50-diameter size, is the first layer. The words are transformed into a vector space model by this layer based on how frequently a word appears next to other words. To learn embedding for each term in the training dataset, the embedding layer employs random weights (Mikolov et al., 2013). The kernel of size (3 x 3) using ReLU as an activation function for feature extraction is introduced to the model along with two convolutional layers, each with 100 filters. A max pooling layer of size (2 x 2) is added to limit the feature map's dimensionality. Finally, using the flatten layer, the feature maps are transformed into single-column vectors. For each, the output is sent to a dense layer of 100 neurons.

The CNN was initially tried with 100 epochs to see its overall performance and to analyse the validation loss and validation accuracy. The plots are as shown in the Figure 25 below. The

confusion matrix was also implemented to determine the ratio of correct predictions vs the incorrect ones as shown in the Figure 26 below.

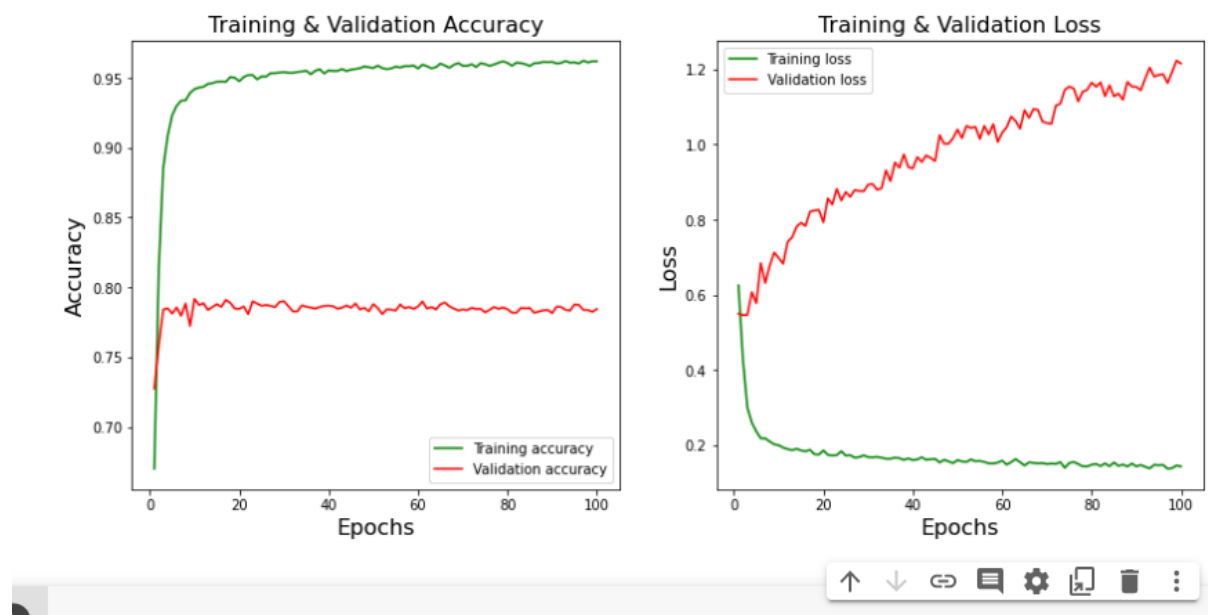


Figure 25: The plots for the validation accuracy and training accuracy alongside the training loss and validation loss before early stopping

From the plots above it is seen that the validation accuracy and the training accuracy remains constant after a certain point of time showing that the learning has stopped, and it is better to implement early stopping to get the final results. The plot after implementing early stopping is as shown in the Figure 27 below. Training loss seems to be decreasing very rapidly in comparison to the validation loss. The results for the precision, recall, accuracy and f1-score has been summarized for all the models in the Table 16 as shown below.

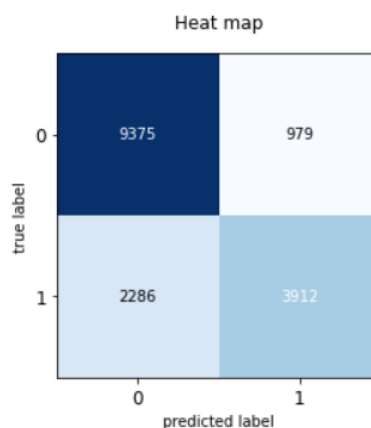


Figure 26: Confusion matrix for CNN

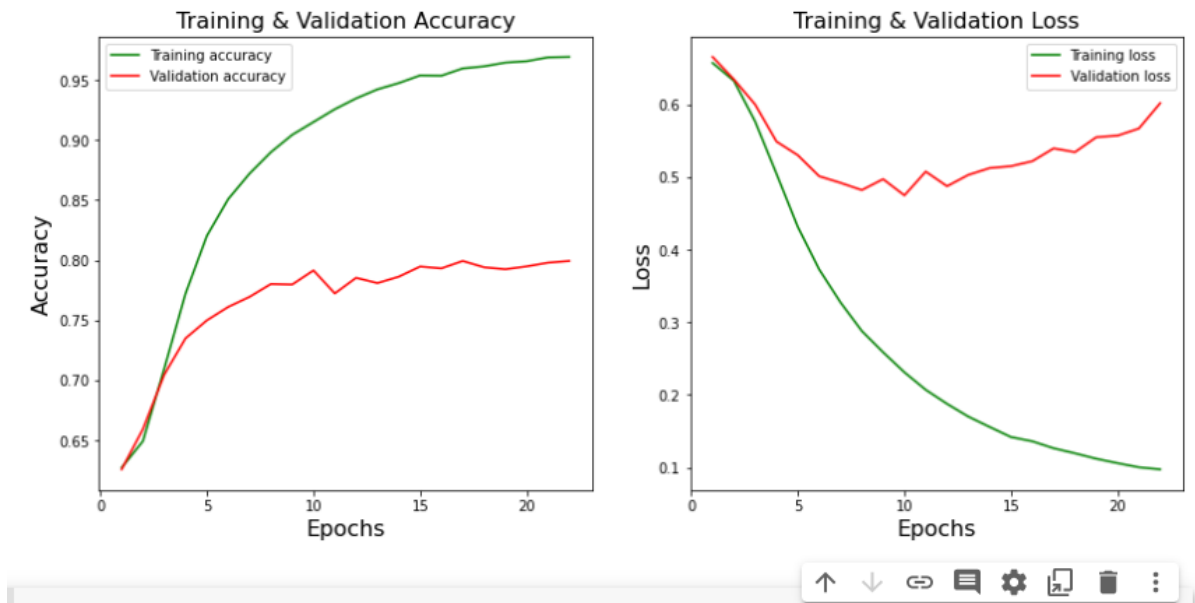


Figure 27: The plots for the validation accuracy and training accuracy alongside the training loss and validation loss after early stopping

All of the implemented models were put to the test by changing the hyper-parameters, such as the kernel function, probability criteria, number of layers, epochs, and embedding dimensions. Different classification criteria, including accuracy, precision, recall, and F1 score, for the deep learning models and AUC-ROC and AUC-PR for the traditional algorithms were used to evaluate each classification model against each other. The True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) were all obtained from the confusion matrix and used to compute the classification metrics. The formulas for each of the metrics are provided below:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \dots \dots \dots (8)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \dots \dots \dots (9)$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \dots \dots \dots (10)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \dots \dots \dots (11)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \dots \dots \dots (12)$$

$$\text{f1 - score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \dots \dots \dots (12)$$

To summarise, the Table 16 gives the comparison of the relevant performance metrics obtained across all the three models mentioned above.

Metrics/Models	ANN	LSTM	CNN
Precision	0.855	0.722	0.800
Recall	0.577	0.703	0.631
F1-score	0.773	0.772	0.779
Accuracy	0.805	0.787	0.803

Table 16: Summary of Performance metrics

11. Conclusion:

The rapid development of genomic technology has opened up new opportunities for medical practitioners and biotechnologists. Understanding the disease's progression can be made easier by identifying the Pca outcomes signature. The advancement of technologies related to biological sciences also presents difficulties. The application of various deep learning models in bioinformatics and medicine is fraught with considerable challenges. Additionally, the interdisciplinary approach has improved models that can be used to solve prediction issues. The dataset, which was tested and subsequently encoded using both label-encoding and k-mer encoding, was taken from the Cosmic Sanger database for cancer research.

11.1 Reflecting on the obtained results:

The supervised traditional machine learning algorithms namely SVM, Naïve Bayes and Logistic Regression were originally not part of the study, they were used as a preliminary to check if the genomic sequence data was in an acceptable format for the application purposes. Then the idea of implementing it for a single layer neural network was added in to treat it as a building block for the design of ANN, LSTM and CNN. The input sequence can be viewed as a long sentence of k-mer words which can be treated as a text classification problem which can parse these words as grammar structures. Even though SVM, Naïve Bayes and logistic regression perform quite well, they have a drawback of not learning any structure since they were only focussed on bag-of-words representation which does not follow any order. In the case of structured learning with ANN, CNN and RNN (LSTM), we not only retain that information but also have access to additional information from the structures, such as learning phrases directly rather than as n-grams. It should be noted that to create an effective model using deep learning networks, we will want far more instances than we typically use for SVM-based models, which is appropriate for our study since a dataset with 55000 samples is regarded as adequate. Moreover, looking at the summary Table 16 it is evident that ANNs outperform the traditional models when compared to other deep learning model. The metrics alone cannot be a complete proof to determine the extent to which the model can be adaptable to a larger dataset. Even though ANN has the best performance overall the point here is that the explorative analysis of CNN definitely showed its application in the classification of textual data rather than its conventional usage as an image classifier.

11.2 Numerically Predominant models:

The study was initially started with building a Naïve Bayes classifier and SVM (both linear and Gaussian RBF kernel). SVM performed exceptionally well when compared to Naïve Bayes after analysing the confusion matrix obtained. Linear SVM outperformed the non-linear one by making 5260 correct predictions for neutral condition and 3260 correct predictions for pathogenic condition. The results obtained were much better when compared to the ones implemented without the stratified 5-fold cross-validation technique. Logistic regression and Perceptron were later added to explore their strengths and predictive power. Performance metrics such as accuracy, AUC-ROC and AUC-PR were calculated. Comparatively LGR had the

best results in terms of AUC-ROC and AUC-PR which were 91.9% and 88.1% respectively, but the Perceptron had better accuracy by 1%. When it comes to the deep learning models ANN had the best accuracy, precision, f1-score and recall rate of 0.805, 0.855, 0.773 and 0.577 respectively.

11.3 Closure:

To the best of my knowledge, the three models represent a breakthrough in the search for hallmark gene biomarkers for Pca progression. The classifiers have performed well based on the available data. There are some assumptions that one should not ignore, especially when it comes to the ratio of samples belonging to white men and black men when it is evident that the aggressiveness and the rapidity of cancer growth is much more intensified in black men comparatively. The data collected from the sanger database is assumed to have taken into consideration the factors such as this to have a fair representation of different races and groups. The deep learning models correctly distinguishes between all malignant and non-cancer samples exploiting the data gleaned in genomic diversity, which may imply that the genetic markers for cancer can be determined in all people without taking into account racial or sexual differences. When it comes to customised medicine, the prediction job is one of the most beneficial tasks. Experts in bioinformatics and medicine can examine different types of gene mutation because of the progress made in the field of DNA sequence prediction using various Deep learning techniques. Given the model's results and the reasonable accuracy of the method, it promises to be an efficient procedure of DNA microarray gene expression.

Currently, the models are built to forecast results based on genomic profiling, including gene expressions and RNA-Seq. The expressions or quantifications of the biomarkers in the blood or urine can be measured utilising pathways databases and wet-lab studies, and based on these measurements, the Pca outcome can be predicted (Alkhateeb, Atikukke and Rueda, 2020). Hyperparameter optimizers that combine hybrid DL-based classifiers with metaheuristics can be created in the future to improve PCa detection outcomes. Overall, an attempt has been made to replicate the current state-of-the-art models and to test their explorative powers.

12. Future Work:

One of the upgrades that can be performed on the presented models in this study is to improve their performance on the validation dataset by optimizing the parameters using hyperparameter tuning. Although care was taken about choosing many parameters like the filter size, kernel size or the learning rate etc., these were not sufficient to exploit the full fledged working of the neural network models.

To categorise prostatic and non-prostatic subjects, we can use CNN and other deep learning techniques together including the basic ANN and LSTM with transfer learning strategies, feature selection, and ranking algorithms. In order to compare the outcomes with the wider dataset, we can also gather the clinical characteristics of the individuals. Although the texture

features for pre-processing were used in this study, we can still use same procedures with other feature extraction techniques in the future. The effectiveness of any DL model can be fully realised by using them together may be with the combination of CNN-LSTM or ANN-LSTM since gradients that vanish and explode on the LSTM are a problem and the interconnected sequence prediction task parameters cannot be clearly correlated by the LSTM model. The features extracted by the convolutional layers can be fed as the input to LSTM which can then be used for the classification task. Bi-directional LSTM is known to perform better than the normal LSTM and can be replaced with the existing model used during the study to form a hybrid model which has a potential of obtaining more accuracy.

Another area of interest peaks from the use of autoencoders and decoders. They are usually used for unsupervised classification applications to learn the data encodings. Like CNN their applications are mostly on the image data, but there is definitely some scope of them being used for the textual data in conjunction with NLP processing tasks to create a sophisticated model. The aim of autoencoders is to learn the lower dimensional representation of a higher dimensional data which seems plausible in the case of capturing the features of a lengthy DNA sequence used during the study.

13. Project Management:

The timeline for the proposed study is as shown in the Table 17. The timelines have almost been maintained as highlighted in the table shown. The time period allocated for each task was almost on track except for few exceptions.

The collection of the raw dataset from the cosmic sanger database was a difficult task since it was needed to be familiar with the GUI of the website and the filter criteria to fetch the genes of interest. The support team from the website were really cooperative and guided through the entire process else it would not have been possible to obtain the datasets used for the study.

Initially all the coding work had been done on my local machine using Jupyter notebook, but after a certain point of time there were severe issues with the downloading of packages and some of the code blocks started to break eventually. In addition to that the results obtained and reported during the interim report writing is much different from the results reported in the final thesis due to the environment issues and the problems with the system GPU. Hence the decision was made to purchase a pro version of Google collab which allowed access to external GPUs and made the saving of files easier. With the resources available on my local machine LSTM took very long period of time for training and the training time did not seem to improve even when used on the google collab pro version. Hence it was only run for 10 epochs.

Task	March	April	May	June	July	August	September
Literature Review/Research							
Data Collection							
Model Development/Training							
Model Testing							
Model Evaluation Review and Result analysis							
Preparing Presentation							
Interim Report							
Dissertation Report							

Table 17:Project Timeline

Bibliography:

- [1] Abass, Y. and Adeshina, S. (2021). Feature Selection with Ensemble Learning for Prostate Cancer Prediction from Gene Expression. *IJCSNS International Journal of Computer Science and Network Security*, [online] 21(12), p.526. doi:10.22937/IJCSNS.2021.21.12.73.
- [2] Ahn, T., Goo, T., Lee, C.H., Kim, S., Han, K., Park, S. and Park, T. (2020). Deep learning-based classification and interpretation of gene expression data from cancer and normal tissues. *International Journal of Data Mining and Bioinformatics*, 24(2), p.121. doi:10.1504/ijdmb.2020.110155.
- [3] Ai, D., Wang, Y., Li, X. and Pan, H. (2020). Colorectal Cancer Prediction Based on Weighted Gene Co-Expression Network Analysis and Variational Auto-Encoder. *Biomolecules*, 10(9), p.1207. doi:10.3390/biom10091207.
- [4] Alkhateeb, A., Atikukke, G. and Rueda, L. (2020). Machine learning methods for prostate cancer diagnosis. *probiologists.com*, [online] 1(3). Available at: <https://probiologists.com/Article/Machine-learning-methods-for-prostate-cancer-diagnosis> [Accessed 4 Sep. 2022].
- [5] Alkhateeb, A., Rezaeian, I., Singireddy, S., Cavallo-Medved, D., Porter, L.A. and Rueda, L. (2019). Transcriptomics Signature from Next-Generation Sequencing Data Reveals New Transcriptomic Biomarkers Related to Prostate Cancer. *Cancer Informatics*, 18, p.117693511983552. doi:10.1177/1176935119835522.
- [6] Amin, M.B., Greene, F.L., Edge, S.B., Compton, C.C., Gershenwald, J.E., Brookland, R.K., Meyer, L., Gress, D.M., Byrd, D.R. and Winchester, D.P. (2017). The Eighth Edition AJCC Cancer Staging Manual: Continuing to build a bridge from a population-based to a more 'personalized' approach to cancer staging. *CA: A Cancer Journal for Clinicians*, 67(2), pp.93–99. doi:10.3322/caac.21388.
- [7] Analytics Vidhya. (2021a). *ANN for Data Science | Basics Of Artificial Neural Network*. [online] Available at: <https://www.analyticsvidhya.com/blog/2021/07/understanding-the-basics-of-artificial-neural-network-ann/> [Accessed 16 Aug. 2022].
- [8] Analytics Vidhya. (2021b). *Artificial Neural Network | Beginners Guide to ANN*. [online] Available at: <https://www.analyticsvidhya.com/blog/2021/05/beginners-guide-to-artificial-neural-network/> [Accessed 17 Aug. 2022].
- [9] Angelini, C., Bongcam-Rudloff, E., Decarli, A., Mv, P. and Stefano, R. (2015). *2th International Meeting on Computational Intelligence Methods for Conference Proceedings*. [online] Available at: <https://www.scedt.tees.ac.uk/c.angione/papers/BookProceedingsCIBB2015.pdf> [Accessed 8 Aug. 2022].
- [10] Arel, I., Rose, D.C. and Karnowski, T.P. (2010). Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]. *IEEE Computational Intelligence Magazine*, 5(4), pp.13–18. doi:10.1109/mci.2010.938364.
- [11] Azizi, S., Bayat, S., Yan, P., Tahmasebi, A., Kwak, J.T., Xu, S., Turkbey, B., Choyke, P., Pinto, P., Wood, B., Mousavi, P. and Abolmaesumi, P. (2018). Deep Recurrent Neural Networks for Prostate Cancer Detection: Analysis of Temporal Enhanced Ultrasound. *IEEE Transactions on Medical Imaging*, 37(12), pp.2695–2703. doi:10.1109/tmi.2018.2849959.

- [12] Bay, N.S.-Y. and Bay, B.-H. (2010). Greek anatomist herophilus: the father of anatomy. *Anatomy & Cell Biology*, 43(4), p.280. doi:10.5115/acb.2010.43.4.280.
- [13] Beer, T.M., Armstrong, A.J., Rathkopf, D.E., Loriot, Y., Sternberg, C.N., Higano, C.S., Iversen, P., Bhattacharya, S., Carles, J., Chowdhury, S., Davis, I.D., de Bono, J.S., Evans, C.P., Fizazi, K., Joshua, A.M., Kim, C.-S., Kimura, G., Mainwaring, P., Mansbach, H. and Miller, K. (2014). Enzalutamide in metastatic prostate cancer before chemotherapy. *The New England journal of medicine*, [online] 371(5), pp.424–33. doi:10.1056/NEJMoa1405095.
- [14] Bertoli, G., Cava, C. and Castiglioni, I. (2016). MicroRNAs as Biomarkers for Diagnosis, Prognosis and Theranostics in Prostate Cancer. *International Journal of Molecular Sciences*, 17(3), p.421. doi:10.3390/ijms17030421.
- [15] Brawley, O.W. (2012). Prostate cancer epidemiology in the United States. *World Journal of Urology*, 30(2), pp.195–200. doi:10.1007/s00345-012-0824-2.
- [16] Bulten, W., Pinckaers, H., van Boven, H., Vink, R., de Bel, T., van Ginneken, B., van der Laak, J., Hulsbergen-van de Kaa, C. and Litjens, G. (2020). Automated deep-learning system for Gleason grading of prostate cancer using biopsies: a diagnostic study. *The Lancet Oncology*, 21(2), pp.233–241. doi:10.1016/s1470-2045(19)30739-9.
- [17] Buyyounouski, M.K., Pickles, T., Kestin, L.L., Allison, R. and Williams, S.G. (2012). Validating the interval to biochemical failure for the identification of potentially lethal prostate cancer. *Journal of clinical oncology*, [online] 30(15), pp.1857–1863. doi:10.1200/jco.2011.35.1924.
- [18] Campanella, G., Silva, V.W.K. and Fuchs, T.J. (2018). Terabyte-scale Deep Multiple Instance Learning for Classification and Localization in Pathology. *ArXiv*. [online] Available at: <https://www.semanticscholar.org/paper/Terabyte-scale-Deep-Multiple-Instance-Learning-for-Campanella-Silva/52df0f3f7d765380370c87faf4e45596bf6c09d7> [Accessed 1 Sep. 2022].
- [19] Chen, P.-H.C., Gadepalli, K., MacDonald, R., Liu, Y., Kadowaki, S., Nagpal, K., Kohlberger, T., Dean, J., Corrado, G.S., Hipp, J.D., Mermel, C.H. and Stumpe, M.C. (2019). An augmented reality microscope with real-time artificial intelligence integration for cancer diagnosis. *Nature Medicine*, 25(9), pp.1453–1457. doi:10.1038/s41591-019-0539-7.
- [20] Chornokur, G., Dalton, K., Borysova, M.E. and Kumar, N.B. (2010). Disparities at presentation, diagnosis, treatment, and survival in African American men, affected by prostate cancer. *The Prostate*, 71(9), pp.985–997. doi:10.1002/pros.21314.
- [21] Cooperberg, M.R. (2013). Re-Examining Racial Disparities in Prostate Cancer Outcomes. *Journal of Clinical Oncology*, 31(24), pp.2979–2980. doi:10.1200/jco.2013.50.7723.
- [22] D’Amico, A.V., Moul, J., Carroll, P.R., Sun, L., Lubeck, D. and Chen, M.-H. (2003). Cancer-Specific Mortality After Surgery or Radiation for Patients With Clinically Localized Prostate Cancer Managed During the Prostate-Specific Antigen Era. *Journal of Clinical Oncology*, 21(11), pp.2163–2172. doi:10.1200/jco.2003.01.075.
- [23] DARENDELI, B.N. and YILMAZ, A. (2021). Convolutional Neural Network Approach to Predict Tumor Samples Using Gene Expression Data. *Journal of Intelligent Systems: Theory and Applications*, 4(2), pp.136–141. doi:10.38016/jista.946954.

- [24]DataFlair. (2017). *Artificial Neural Networks for Machine Learning - Every aspect you need to know about*. [online] Available at: <https://data-flair.training/blogs/artificial-neural-networks-for-machine-learning/> [Accessed 16 Aug. 2022].
- [25]de Bono, J.S., Logothetis, C.J., Molina, A., Fizazi, K., North, S., Chu, L., Chi, K.N., Jones, R.J., Goodman, O.B., Saad, F., Staffurth, J.N., Mainwaring, P., Harland, S., Flaig, T.W., Hutson, T.E., Cheng, T., Patterson, H., Hainsworth, J.D., Ryan, C.J. and Sternberg, C.N. (2011). Abiraterone and Increased Survival in Metastatic Prostate Cancer. *New England Journal of Medicine*, 364(21), pp.1995–2005. doi:10.1056/nejmoa1014618.
- [26]Denmeade, S.R. and Isaacs, J.T. (2002). A history of prostate cancer treatment. *Nature Reviews Cancer*, [online] 2(5), pp.389–396. doi:10.1038/nrc801.
- [27]Desai, H.P., Parameshwaran, A.P., Sunderraman, R. and Weeks, M. (2020). Comparative Study Using Neural Networks for 16S Ribosomal Gene Classification. *Journal of Computational Biology*, 27(2), pp.248–258. doi:10.1089/cmb.2019.0436.
- [28]DeSantis, C.E., Siegel, R.L., Sauer, A.G., Miller, K.D., Fedewa, S.A., Alcaraz, K.I. and Jemal, A. (2016). Cancer statistics for African Americans, 2016: Progress and opportunities in reducing racial disparities. *CA: A Cancer Journal for Clinicians*, 66(4), pp.290–308. doi:10.3322/caac.21340.
- [29]Edgar, R.C. (2010). Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, 26(19), pp.2460–2461. doi:10.1093/bioinformatics/btq461.
- [30]Edge, S.B. and Compton, C.C. (2010). The American Joint Committee on Cancer: the 7th edition of the AJCC cancer staging manual and the future of TNM. *Annals of surgical oncology*, [online] 17(6), pp.1471–4. doi:10.1245/s10434-010-0985-4.
- [31]Elmarakeby, H.A., Hwang, J., Arafeh, R., Crowdis, J., Gang, S., Liu, D., AlDubayan, S.H., Salari, K., Kregel, S., Richter, C., Arnoff, T.E., Park, J., Hahn, W.C. and Van Allen, E.M. (2021). Biologically informed deep neural network for prostate cancer discovery. *Nature*, [online] 598(7880), pp.348–352. doi:10.1038/s41586-021-03922-4.
- [32]Epstein, J.I. (2010). An Update of the Gleason Grading System. *Journal of Urology*, 183(2), pp.433–440. doi:10.1016/j.juro.2009.10.046.
- [33]Evans, S., Metcalfe, C., Ibrahim, F., Persad, R. and Ben-Shlomo, Y. (2008). Investigating Black-White differences in prostate cancer prognosis: A systematic review and meta-analysis. *International Journal of Cancer*, 123(2), pp.430–435. doi:10.1002/ijc.23500.
- [34]Feng, C. (2022). *LSTM Recurrent Neural Network - Machine Learning Notebook*. [online] Gitbook.io. Available at: https://calvinfeng.gitbook.io/machine-learning-notebook/supervised-learning/recurrent-neural-network/long_short_term_memory [Accessed 8 Aug. 2022].
- [35]Feng, Y., Yang, F., Zhou, X., Guo, Y., Tang, F., Ren, F., Guo, J. and Ji, S. (2019). A Deep Learning Approach for Targeted Contrast-Enhanced Ultrasound Based Prostate Cancer Detection. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 16(6), pp.1794–1801. doi:10.1109/tcbb.2018.2835444.
- [36]Fieres, J., Schemmel, J. and Meier, K. (2006). *Training convolutional networks of threshold neurons suited for low-power hardware implementation*. [online] Available at: http://www.kip.uni-heidelberg.de/Veroeffentlichungen/download.cgi/4606/ps/HD-KIP_06-21.pdf [Accessed 17 Aug. 2022].

- [37]Glaab, E., Bacardit, J., Garibaldi, J.M. and Krasnogor, N. (2012). Using Rule-Based Machine Learning for Candidate Disease Gene Prioritization and Sample Classification of Cancer Gene Expression Data. *PLoS ONE*, 7(7), p.e39932. doi:10.1371/journal.pone.0039932.
- [38]Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*. [online] Deeplearningbook.org. Available at: <https://www.deeplearningbook.org/>.
- [39]Gunasekaran, H., Ramalakshmi, K., Rex Macedo Arokiaraj, A., Deepa Kanmani, S., Venkatesan, C. and Suresh Gnana Dhas, C. (2021). Analysis of DNA Sequence Classification Using CNN and Hybrid Models. *Computational and Mathematical Methods in Medicine*, 2021, pp.1–12. doi:10.1155/2021/1835056.
- [40]Guo, H., Nguyen, H., Vu, D.-A. and Bui, X.-N. (2019). Forecasting mining capital cost for open-pit mining projects based on artificial neural network approach. *Resources Policy*, p.101474. doi:10.1016/j.resourpol.2019.101474.
- [41]Hamzeh, O., Alkhateeb, A., Rezaeian, I., Karkar, A. and Rueda, L. (2017). Finding Transcripts Associated with Prostate Cancer Gleason Stages Using Next Generation Sequencing and Machine Learning Techniques. *Bioinformatics and Biomedical Engineering*, pp.337–348. doi:10.1007/978-3-319-56154-7_31.
- [42]Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), pp.1735–1780. doi:10.1162/neco.1997.9.8.1735.
- [43]Huisman, A., Looijen, A., van den Brink, S.M. and van Diest, P.J. (2010). Creation of a fully digital pathology slide archive by high-volume tissue slide scanning. *Human Pathology*, 41(5), pp.751–757. doi:10.1016/j.humpath.2009.08.026.
- [44]Huo, Y., Xin, L., Kang, C., Wang, M., Ma, Q. and Yu, B. (2020). SGL-SVM: A novel method for tumor classification via support vector machine with sparse group Lasso. *Journal of Theoretical Biology*, [online] 486, p.110098. doi:10.1016/j.jtbi.2019.110098.
- [45]Iqbal, S., Siddiqui, G.F., Rehman, A., Hussain, L., Saba, T., Tariq, U. and Abbasi, A.A. (2021). Prostate Cancer Detection Using Deep Learning and Traditional Techniques. *IEEE Access*, 9, pp.27085–27100. doi:10.1109/access.2021.3057654.
- [46]Jahnavi Mahanta (2017). *Introduction to Neural Networks, Advantages and Applications*. [online] Medium. Available at: <https://towardsdatascience.com/introduction-to-neural-networks-advantages-and-applications-96851bd1a207>.
- [47]Josef Marx, F. and Karenberg, A. (2009). History of the Term Prostate. *The Prostate*, 69(2), pp.208–213. doi:10.1002/pros.20871.
- [48]K, N., Rajaguru, H. and P, R. (2021). *Microarray Prostate Cancer Classification using Eminent Genes*. [online] IEEE Xplore. doi:10.1109/STCR51658.2021.9588811.
- [49]Kang, C., Huo, Y., Xin, L., Tian, B. and Yu, B. (2019). Feature selection and tumor classification for microarray data using relaxed Lasso and generalized multi-class support vector machine. *Journal of Theoretical Biology*, 463, pp.77–91. doi:10.1016/j.jtbi.2018.12.010.
- [50]Kattan, M.W., Eastham, J.A., Stapleton, A.M., Wheeler, T.M. and Scardino, P.T. (1998). A preoperative nomogram for disease recurrence following radical prostatectomy for prostate

- cancer. *Journal of the National Cancer Institute*, [online] 90(10), pp.766–771. doi:10.1093/jnci/90.10.766.
- [51]Kumar, R., Bhanti, P., Marwal, A. and Gaur, R.K. (2019). Gene Expression-Based Supervised Classification Models for Discriminating Early- and Late-Stage Prostate Cancer. *Proceedings of the National Academy of Sciences, India Section B: Biological Sciences*, 90(3), pp.541–565. doi:10.1007/s40011-019-01127-4.
- [52]Kwak, J.T. and Hewitt, S.M. (2017). Lumen-based detection of prostate cancer via convolutional neural networks. *NASA ADS*, [online] 10140, p.1014008. doi:10.1117/12.2253513.
- [53]LANCHANTIN, J., SINGH, R., WANG, B. and QI, Y. (2016). DEEP MOTIF DASHBOARD: VISUALIZING AND UNDERSTANDING GENOMIC SEQUENCES USING DEEP NEURAL NETWORKS. *Biocomputing 2017*. [online] doi:10.1142/9789813207813_0025.
- [54]Lecun, Y., L Eon Bottou, Bengio, Y. and Patrick Haaner Abstract | (2006). Gradient-Based Learning Applied to Document Recognition. *PROC. OF THE IEEE*, [online] 86(11). Available at: http://vision.stanford.edu/cs598_spring07/papers/Lecun98.pdf.
- [55]Lee, B., Baek, J., Park, S. and Yoon, S. (2016). deepTarget. *Proceedings of the 7th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. doi:10.1145/2975167.2975212.
- [56]Lee, K.B., Cheon, S. and Kim, C.O. (2017). A Convolutional Neural Network for Fault Classification and Diagnosis in Semiconductor Manufacturing Processes. *IEEE Transactions on Semiconductor Manufacturing*, [online] 30(2), pp.135–142. doi:10.1109/tsm.2017.2676245.
- [57]Lee, S., Kerns, S., Ostrer, H., Rosenstein, B., Deasy, J.O. and Oh, J.H. (2018). Machine Learning on a Genome-wide Association Study to Predict Late Genitourinary Toxicity After Prostate Radiation Therapy. *International Journal of Radiation Oncology*Biophysics*, 101(1), pp.128–135. doi:10.1016/j.ijrobp.2018.01.054.
- [58]Lenain, R., Seneviratne, M.G., Bozkurt, S., Blayney, D.W., Brooks, J.D. and Hernandez-Boussard, T. (2019). Machine Learning Approaches for Extracting Stage from Pathology Reports in Prostate Cancer. *Studies in Health Technology and Informatics*, [online] 264, pp.1522–1523. doi:10.3233/SHTI190515.
- [59]Lewis, B.P., Shih, I-hung., Jones-Rhoades, M.W., Bartel, D.P. and Burge, C.B. (2003). Prediction of Mammalian MicroRNA Targets. *Cell*, [online] 115(7), pp.787–798. doi:10.1016/s0092-8674(03)01018-3.
- [60]Li, J., Cheng, J., Shi, J. and Huang, F. (2012a). Brief Introduction of Back Propagation (BP) Neural Network Algorithm and Its Improvement. *Advances in Intelligent and Soft Computing*, 169, pp.553–558. doi:10.1007/978-3-642-30223-7_87.
- [61]Li, Y., Chan, S.C., Brand, L.J., Hwang, T.H., Silverstein, K.A.T. and Dehm, S.M. (2012b). Androgen Receptor Splice Variants Mediate Enzalutamide Resistance in Castration-Resistant Prostate Cancer Cell Lines. *Cancer Research*, 73(2), pp.483–489. doi:10.1158/0008-5472.can-12-3630.
- [62]Litjens, G., Sánchez, C.I., Timofeeva, N., Hermsen, M., Nagtegaal, I., Kovacs, I., Hulsbergen - van de Kaa, C., Bult, P., van Ginneken, B. and van der Laak, J. (2016). Deep learning as a tool for increased accuracy and efficiency of histopathological diagnosis. *Scientific Reports*, [online] 6(1). doi:10.1038/srep26286.

- [63]Liu, X., Krishnan, A. and Mondry, A. (2005). An entropy-based gene selection method for cancer classification using microarray data. *BMC Bioinformatics*, 6(1), p.76. doi:10.1186/1471-2105-6-76.
- [64]Liu, Z., Tang, D., Cai, Y., Wang, R. and Chen, F. (2017). A hybrid method based on ensemble WELM for handling multi class imbalance in cancer microarray data. *Neurocomputing*, 266, pp.641–650. doi:10.1016/j.neucom.2017.05.066.
- [65]Luca, B.-A., Moulton, V., Ellis, C., Edwards, D.R., Campbell, C., Cooper, R.A., Clark, J., Brewer, D.S. and Cooper, C.S. (2020). A novel stratification framework for predicting outcome in patients with prostate cancer. *British Journal of Cancer*, [online] 122(10), pp.1467–1476. doi:10.1038/s41416-020-0799-5.
- [66]Lucas, M., Jansen, I., Savci-Heijink, C.D., Meijer, S.L., de Boer, O.J., van Leeuwen, T.G., de Bruin, D.M. and Marquering, H.A. (2019). Deep learning for automatic Gleason pattern classification for grade group determination of prostate biopsies. *Virchows Archiv*, 475(1), pp.77–83. doi:10.1007/s00428-019-02577-x.
- [67]Lynch, H.T., Kosoko-Lasaki, O., Leslie, S.W., Rendell, M., Shaw, T., Snyder, C., D’Amico, A.V., Buxbaum, S., Isaacs, W.B., Loeb, S., Moul, J.W. and Powell, I. (2016). Screening for familial and hereditary prostate cancer. *International Journal of Cancer*, 138(11), pp.2579–2591. doi:10.1002/ijc.29949.
- [68]MacInnis, R.J., Schmidt, D.F., Makalic, E., Severi, G., FitzGerald, L.M., Reumann, M., Kapuscinski, M.K., Kowalczyk, A., Zhou, Z., Goudey, B., Qian, G., Bui, Q.M., Park, D.J., Freeman, A., Southey, M.C., Al Olama, A.A., Kote-Jarai, Z., Eeles, R.A., Hopper, J.L. and Giles, G.G. (2016). Use of a Novel Nonparametric Version of DEPTH to Identify Genomic Regions Associated with Prostate Cancer Risk. *Cancer Epidemiology Biomarkers & Prevention*, 25(12), pp.1619–1624. doi:10.1158/1055-9965.epi-16-0301.
- [69]Marchiori, E. and Sebag, M. (2005). Bayesian Learning with Local Support Vector Machines for Cancer Classification with Gene Expression Data. *Lecture Notes in Computer Science*, 3449, pp.74–83. doi:10.1007/978-3-540-32003-6_8.
- [70]Martin, D.N., Starks, A.M. and Ambis, S. (2013). Biological determinants of health disparities in prostate cancer. *Current Opinion in Oncology*, p.1. doi:10.1097/cco.0b013e32835eb5d1.
- [71]Mathur, M. (2018). Bioinformatics challenges: A review. *International Journal of Advanced Scientific Research*, [online] 3(6), pp.29–33. Available at: <http://www.allscientificjournal.com/archives/2018/vol3/issue6/3-6-22>.
- [72]McDermott, M.B.A., Wang, J., Zhao, W.-N., Sheridan, S.D., Szolovits, P., Kohane, I., Haggarty, S.J. and Perlis, R.H. (2020). Deep Learning Benchmarks on L1000 Gene Expression Data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, [online] 17(6), pp.1846–1857. doi:10.1109/tcbb.2019.2910061.
- [73]Mellinger, G.T., Gleason, D. and Bailer, J. (1967). The histology and prognosis of prostatic cancer. *The Journal of Urology*, [online] 97(2), pp.331–337. doi:10.1016/s0022-5347(17)63039-8.
- [74]Melville, A. (1984). Set and serendipity in the detection of drug hazards. *Social science & medicine* (1982), [online] 19(4), pp.391–396. doi:10.1016/0277-9536(84)90196-5.
- [75]Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space*. [online] arXiv.org. Available at: <https://arxiv.org/abs/1301.3781>.

- [76] Mohapatra, P., Chakravarty, S. and Dash, P.K. (2016). Microarray medical data classification using kernel ridge regression and modified cat swarm optimization based gene selection system. *Swarm and Evolutionary Computation*, 28, pp.144–160. doi:10.1016/j.swevo.2016.02.002.
- [77] Momenzadeh, M., Sehhati, M. and Rabbani, H. (2019). A novel feature selection method for microarray data classification based on hidden Markov model. *Journal of Biomedical Informatics*, 95, p.103213. doi:10.1016/j.jbi.2019.103213.
- [78] Nagpal, K., Foote, D., Liu, Y., Chen, P.-H.C., Wulczyn, E., Tan, F., Olson, N., Smith, J.L., Mohtashamian, A., Wren, J.H., Corrado, G.S., MacDonald, R., Peng, L.H., Amin, M.B., Evans, A.J., Sangoi, A.R., Mermel, C.H., Hipp, J.D. and Stumpe, M.C. (2019). Development and validation of a deep learning algorithm for improving Gleason scoring of prostate cancer. *npj Digital Medicine*, 2(1). doi:10.1038/s41746-019-0112-2.
- [79] Nebauer, C. (1998). Evaluation of convolutional neural networks for visual recognition. *IEEE Transactions on Neural Networks*, 9(4), pp.685–696. doi:10.1109/72.701181.
- [80] Nguyen, N.G., Tran, V.A., Phan, D., Lumbanraja, F.R., Faisal, M.R., Abapihi, B., Kubo, M. and Satou, K. (2016). DNA Sequence Classification by Convolutional Neural Network. *Journal Biomedical Science and Engineering*, [online] 9(5), pp.280–286. Available at: <http://repository.lppm.unila.ac.id/28063/> [Accessed 18 Aug. 2022].
- [81] Nir, G., Hor, S., Karimi, D., Fazli, L., Skinnider, B.F., Tavassoli, P., Turbin, D., Villamil, C.F., Wang, G., Wilson, R.S., Iczkowski, K.A., Lucia, M.S., Black, P.C., Abolmaesumi, P., Goldenberg, S.L. and Salcudean, S.E. (2018). Automatic grading of prostate cancer in digitized histopathology images: Learning from multiple experts. *Medical Image Analysis*, 50, pp.167–180. doi:10.1016/j.media.2018.09.005.
- [82] Otori, M., Wheeler, T.M. and Scardino, P.T. (1994). The New American Joint Committee on Cancer and International Union Against Cancer TNM classification of prostate cancer. Clinicopathologic correlations. *Cancer*, [online] 74(1), pp.104–114. doi:10.1002/1097-0142(19940701)74:13.0.co;2-5.
- [83] Olah, C. (2015). *Understanding LSTM Networks -- colah's blog*. [online] Github.io. Available at: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> [Accessed 8 Aug. 2022].
- [84] Olender, J. and Lee, N.H. (2019). Role of Alternative Splicing in Prostate Cancer Aggressiveness and Drug Resistance in African Americans. *Advances in Experimental Medicine and Biology*, [online] pp.119–139. doi:10.1007/978-3-030-22254-3_10.
- [85] Olson, J.S. (1989a). *The history of cancer : an annotated bibliography*. Westport, Conn.: Greenwood Press.
- [86] Olson, J.S. (1989b). *The History of Cancer: An Annotated Bibliography*. [online] Google Books. ABC-CLIO. Available at: https://books.google.co.uk/books?hl=en&lr=&id=oAX8jOJ9uO0C&oi=fnd&pg=PR5&ots=0YsaM8rEq-&sig=74yyvpSevKfokNIFlokjTI3JL5k&redir_esc=y#v=onepage&q&f=false [Accessed 7 Aug. 2022].
- [87] Oltean, S., Sorg, B.S., Albrecht, T., Bonano, V.I., Brazas, R.M., Dewhirst, M.W. and Garcia-Blanco, M.A. (2006). Alternative inclusion of fibroblast growth factor receptor 2 exon IIIc in Dunning prostate tumors reveals unexpected epithelial mesenchymal plasticity. *Proceedings of the*

- National Academy of Sciences of the United States of America*, [online] 103(38), pp.14116–14121. doi:10.1073/pnas.0603090103.
- [88]Pan, Q., Shai, O., Lee, L.J., Frey, B.J. and Blencowe, B.J. (2008). Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nature genetics*, [online] 40(12), pp.1413–5. doi:10.1038/ng.259.
- [89]Pantanowitz, L. (2010). Digital images and the future of digital pathology. *Journal of Pathology Informatics*, 1(1), p.15. doi:10.4103/2153-3539.68332.
- [90]Papsidero, L.D., Wang, M.C., Valenzuela, L.A., Murphy, G.P. and Chu, T.M. (1980). A prostate antigen in sera of prostatic cancer patients. *Cancer Research*, [online] 40(7), pp.2428–2432. Available at: <https://pubmed.ncbi.nlm.nih.gov/7388802/>.
- [91]Park, S., Min, S., Choi, H.-S. and Yoon, S. (2016). deepMiRGene: Deep Neural Network based Precursor microRNA Prediction. *ArXiv*. [online] Available at: <https://www.semanticscholar.org/paper/deepMiRGene%3A-Deep-Neural-Network-based-Precursor-Park-Min/9d9c573ca0da8daf06fb2627e792d23e59da4bb4> [Accessed 8 Aug. 2022].
- [92]Parker, C., Nilsson, S., Heinrich, D., Helle, S.I., O’Sullivan, J.M., Fosså, S.D., Chodacki, A., Wiechno, P., Logue, J., Seke, M., Widmark, A., Johannessen, D.C., Hoskin, P., Bottomley, D., James, N.D., Solberg, A., Syndikus, I., Kliment, J., Wedel, S. and Boehmer, S. (2013). Alpha Emitter Radium-223 and Survival in Metastatic Prostate Cancer. *New England Journal of Medicine*, 369(3), pp.213–223. doi:10.1056/nejmoa1213755.
- [93]Pernar, C.H., Ebot, E.M., Wilson, K.M. and Mucci, L.A. (2018). The Epidemiology of Prostate Cancer. *Cold Spring Harbor Perspectives in Medicine*, 8(12), p.a030361. doi:10.1101/cshperspect.a030361.
- [94]Pinello, L., Lo Bosco, G. and Yuan, G.-C. . (2013). Applications of alignment-free methods in epigenomics. *Briefings in Bioinformatics*, 15(3), pp.419–430. doi:10.1093/bib/bbt078.
- [95]Powell, I.J. (2007). Epidemiology and Pathophysiology of Prostate Cancer in African-American Men. *Journal of Urology*, 177(2), pp.444–449. doi:10.1016/j.juro.2006.09.024.
- [96]Powell, I.J. and Bollig-Fischer, A. (2013). Minireview: The Molecular and Genomic Basis for Prostate Cancer Health Disparities. *Molecular Endocrinology*, [online] 27(6), pp.879–891. doi:10.1210/me.2013-1039.
- [97]Punnen, S., Cooperberg, M.R., D’Amico, A.V., Karakiewicz, P.I., Moul, J.W., Scher, H.I., Schlomm, T. and Freedland, S.J. (2013). Management of biochemical recurrence after primary treatment of prostate cancer: a systematic review of the literature. *European Urology*, [online] 64(6), pp.905–915. doi:10.1016/j.eururo.2013.05.025.
- [98]Robbins, A.S., Whittemore, A.S. and Thom, D.H. (2000). Differences in Socioeconomic Status and Survival among White and Black Men with Prostate Cancer. *American Journal of Epidemiology*, 151(4), pp.409–416. doi:10.1093/oxfordjournals.aje.a010221.
- [99]Roffman, D.A., Hart, G.R., Leapman, M.S., Yu, J.B., Guo, F.L., Ali, I. and Deng, J. (2018). Development and Validation of a Multiparameterized Artificial Neural Network for Prostate Cancer Risk Prediction and Stratification. *JCO clinical cancer informatics*, [online] 2, pp.1–10. doi:10.1200/CCI.17.00119.

- [100] Ryan, C.J., Smith, M.R., de Bono, J.S., Molina, A., Logothetis, C.J., de Souza, P., Fizazi, K., Mainwaring, P., Piulats, J.M., Ng, S., Carles, J., Mulders, P.F.A., Basch, E., Small, E.J., Saad, F., Schrijvers, D., Van Poppel, H., Mukherjee, S.D., Suttman, H. and Gerritsen, W.R. (2013). Abiraterone in Metastatic Prostate Cancer without Previous Chemotherapy. *New England Journal of Medicine*, [online] 368(2), pp.138–148. doi:10.1056/nejmoa1209096.
- [101] S A, K., D, A., A, R. and A F, H. (2019). Classification of genetic expression in prostate cancer using support vector machine method. *Journal of Physics: Conference Series*, 1613.
- [102] Salinas, C.A., Tsodikov, A., Ishak-Howard, M. and Cooney, K.A. (2014). Prostate Cancer in Young Men: An Important Clinical Entity. *Nature reviews. Urology*, [online] 11(6), pp.317–323. doi:10.1038/nrurol.2014.91.
- [103] Scher, H.I., Fizazi, K., Saad, F., Taplin, M.-E., Sternberg, C.N., Miller, K., de Wit, R., Mulders, P., Chi, K.N., Shore, N.D., Armstrong, A.J., Flaig, T.W., Fléchon, A., Mainwaring, P., Fleming, M., Hainsworth, J.D., Hirmand, M., Selby, B., Seely, L. and de Bono, J.S. (2012). Increased Survival with Enzalutamide in Prostate Cancer after Chemotherapy. *New England Journal of Medicine*, [online] 367(13), pp.1187–1197. doi:10.1056/nejmoa1207506.
- [104] Sharma, S., Zapatero-Rodríguez, J. and O’Kennedy, R. (2017). Prostate cancer diagnostics: Clinical challenges and the ongoing need for disruptive and effective diagnostic tools. *Biotechnology Advances*, [online] 35(2), pp.135–149. doi:10.1016/j.biotechadv.2016.11.009.
- [105] Shen, M.M. and Abate-Shen, C. (2010). Molecular genetics of prostate cancer: new prospects for old challenges. *Genes & Development*, 24(18), pp.1967–2000. doi:10.1101/gad.1965810.
- [106] Siegel, R.L., Miller, K.D. and Jemal, A. (2017). Cancer statistics, 2017. *CA: A Cancer Journal for Clinicians*, 67(1), pp.7–30. doi:10.3322/caac.21387.
- [107] Siegel, R.L., Miller, K.D. and Jemal, A. (2018). Cancer statistics, 2018. *CA: A Cancer Journal for Clinicians*, 68(1), pp.7–30. doi:10.3322/caac.21442.
- [108] Silberstein, J.L., Pal, S.K., Lewis, B. and Sartor, O. (2013). Current clinical challenges in prostate cancer. *Translational andrology and urology*, [online] 2(3), pp.122–36. doi:10.3978/j.issn.2223-4683.2013.09.03.
- [109] Singh, R., Lanchantin, J., Robins, G. and Qi, Y. (2016). DeepChrome: deep-learning for predicting gene expression from histone modifications. *Bioinformatics*, 32(17), pp.i639–i648. doi:10.1093/bioinformatics/btw427.
- [110] Smirnov, E.A., Timoshenko, D.M. and Andrianov, S.N. (2014). Comparison of Regularization Methods for ImageNet Classification with Deep Convolutional Neural Networks. *AASRI Procedia*, 6, pp.89–94. doi:10.1016/j.aasri.2014.05.013.
- [111] Tavasoli, N., Rezaee, K., Momenzadeh, M. and Sehhati, M. (2021). An ensemble soft weighted gene selection-based approach and cancer classification using modified metaheuristic learning. *Journal of Computational Design and Engineering*, [online] 8(4), pp.1172–1189. doi:https://doi.org/10.1093/jcde/qwab039.
- [112] Techopedia.com. (2019). *What is an Artificial Neural Network (ANN)? - Definition from Techopedia*. [online] Available at: <https://www.techopedia.com/definition/5967/artificial-neural-network-ann> [Accessed 16 Aug. 2022].

- [113] Tyson, M.D. and Castle, E.P. (2014). Racial disparities in survival for patients with clinically localized prostate cancer adjusted for treatment effects. *Mayo Clinic Proceedings*, [online] 89(3), pp.300–307. doi:10.1016/j.mayocp.2013.11.001.
- [114] Uzma, Al-Obeidat, F., Tubaishat, A., Shah, B. and Halim, Z. (2020). Gene encoder: a feature selection technique through unsupervised deep learning-based clustering for large gene expression data. *Neural Computing and Applications*, 34(11). doi:10.1007/s00521-020-05101-4.
- [115] Wang, Y., Bernhardt, A.J., Cruz, C., Kraus, J.J., Nacson, J., Nicolas, E., Peri, S., van der Gulden, H., van der Heijden, I., O'Brien, S.W., Zhang, Y., Harrell, M.I., Johnson, S.F., Candido Dos Reis, F.J., Pharoah, P.D.P., Karlan, B., Gourley, C., Lambrechts, D., Chenevix-Trench, G. and Olsson, H. (2016). The BRCA1- 11q Alternative Splice Isoform Bypasses Germline Mutations and Promotes Therapeutic Resistance to PARP Inhibition and Cisplatin. *Cancer Research*, [online] 76(9), pp.2778–2790. doi:10.1158/0008-5472.can-16-0186.
- [116] Weiner, A.B., Matulewicz, R.S., Eggener, S.E. and Schaeffer, E.M. (2016). Increasing incidence of metastatic prostate cancer in the United States (2004–2013). *Prostate Cancer and Prostatic Diseases*, 19(4), pp.395–397. doi:10.1038/pcan.2016.30.
- [117] Wu, X., Wang, H.-Y., Shi, P., Sun, R., Wang, X., Luo, Z., Zeng, F., Lebowitz, M.S., Lin, W.-Y., Lu, J.-J., Scherer, R., Price, O., Wang, Z., Zhou, J. and Wang, Y. (2022). Long short-term memory model – A deep learning approach for medical data with irregularity in cancer predication with tumor markers. *Computers in Biology and Medicine*, [online] 144, p.105362. doi:10.1016/j.compbiomed.2022.105362.
- [118] Yu, Y.P., Landsittel, D., Jing, L., Nelson, J., Ren, B., Liu, L., McDonald, C., Thomas, R., Dhir, R., Finkelstein, S., Michalopoulos, G., Becich, M. and Luo, J.-H. (2004). Gene expression alterations in prostate cancer predicting tumor aggression and preceding development of malignancy. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology*, [online] 22(14), pp.2790–2799. doi:10.1200/JCO.2004.05.158.
- [119] Yue, T. and Wang, H. (2018). Deep Learning for Genomics: A Concise Overview. *arXiv:1802.00810 [cs, q-bio]*. [online] Available at: <https://arxiv.org/abs/1802.00810#:~:text=Advancements%20in%20genomic%20research%20such> [Accessed 8 Aug. 2022].
- [120] Zhang, J.X., Yordanov, B., Gaunt, A., Wang, M.X., Dai, P., Chen, Y.-J., Zhang, K., Fang, J.Z., Dalchau, N., Li, J., Phillips, A. and Zhang, D.Y. (2021). A deep learning model for predicting next-generation sequencing depth from DNA sequence. *Nature Communications*, 12(1). doi:10.1038/s41467-021-24497-8.
- [121] Zhang, Z. (2016). *Derivation of Backpropagation in Convolutional Neural Network (CNN)*. [online] Available at: [https://zzutk.github.io/docs/reports/2016.10%20-%20Derivation%20of%20Backpropagation%20in%20Convolutional%20Neural%20Network%20\(CNN\).pdf](https://zzutk.github.io/docs/reports/2016.10%20-%20Derivation%20of%20Backpropagation%20in%20Convolutional%20Neural%20Network%20(CNN).pdf).
- [122] Zhao, J., Chang, L., Gu, X., Liu, J., Sun, B. and Wei, X. (2020). Systematic profiling of alternative splicing signature reveals prognostic predictor for prostate cancer. *Cancer Science*, [online] 111(8), pp.3020–3031. doi:10.1111/cas.14525.
- [123] Zhou, Y., Wang, H., Xu, F. and Jin, Y.-Q. (2016). Polarimetric SAR Image Classification Using Deep Convolutional Neural Networks. *IEEE Geoscience and Remote Sensing Letters*, [online] 13(12), pp.1935–1939. doi:10.1109/lgrs.2016.2618840.

Appendices:

Graphs obtained during stratified 4-fold cross validation:

