JULY 2022

M.Sc. in Electronics and Computer Engineering

MELİH YAYLA

T.C.

HASAN KALYONCU UNIVERSITY

GRADUATE SCHOOL OF

NATURAL AND APPLIED SCIENCES

INTEGRATION OF DEEP LEARNING METHODS IN THE CLASSIFICATION OF RNA-SEQ DATA

M.Sc. THESIS

IN

ELECTRONICS AND COMPUTER ENGINEERING

BY

MELİH YAYLA

JULY 2022

Integration of Deep Learning Methods in the Classification of RNA-SEQ Data

M.Sc. Thesis

In

Electronics And Computer Engineering

Hasan Kalyoncu University

Supervisor

Asst. Prof. Dr. Bülent HAZNEDAR

Melih YAYLA

July 2022

	GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES INSTITUTE M.Sc. ACCEPTANCE AND APPROVAL FROM

Electronics-Computer Engineering M.Sc. (Master of Science) program student **Melih YAYLA** prepared and submitted the thesis titled **"Integration of Deep Learning Methods in the Classification of RNA-SEQ Data"** defended successfully on the date of 05/07/2022 and accepted by the jury as a M.Sc. thesis.

Position	**Title, Name, and Surname** **Department/University**	**Signature:**
Supervisor	Assist. Prof. Dr. Bülent HAZNEDAR Computer Engineering Department Gaziantep University	
Jury Head	Prof. Dr. M. Fatih HASOĞLU Computer Engineering Department Hasan Kalyoncu University	
Jury Member	Assist. Prof. Dr. Mustafa BIÇAKÇI Computer Engineering Department Hasan Kalyoncu University	

This thesis is accepted by the jury members selected by the institute management board and approved by the institute management board.

Prof. Dr. İbrahim Halil GÜZELBEY

Director

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Melih YAYLA

ABSTRACT

INTEGRATION OF DEEP LEARNING METHODS IN THE CLASSIFICATION OF RNA-SEQ DATA

Yayla, Melih

M.Sc. in Electronic Computer Engineering

Supervisor: Asst. Prof. Dr. Bülent HAZNEDAR

July 2022

103 pages

Cervical cancer is a type of gynecological cancer that affects the cells in the lower section of the uterus called the cervix. Cells become abnormal as a result of gene mutations, and uncontrolled division is the primary cause of cancer. On the other hand, Alzheimer's disease is an irreversible neurological disease that causes memory loss and dementia, primarily as a result of brain cell death as people age. The parts of the brain that control thinking, learning, and memory have been injured or destroyed, causing symptoms. Cervical cancer and Alzheimer's disease are both genetic disorders. As a matter of fact, gene expression is significant in the diagnosis and classification of Cervical Cancer and Alzheimer's disease. Many genes' information is kept in RNA-Seq data. To accelerate this approach and assist clinicians in the diagnosis process, methodologies can be constructed using classification algorithms with decreasing the number of irrelevant genes. The objective of this thesis is to use statistical and deep learning approaches to examine Cervical Cancer and Alzheimer's Disease utilizing RNA-Seq datasets built with genes obtained from real samples. To lower the size of the RNA-Seq data set, gene selection is done by 5%, 10%, and 30% among all genes for both datasets, and the gene expression data were generated for each gene according to the importance level of the genes. In three scenarios, the selected genes are trained and tested in the categorization process. Deep Neural networks, convolutional neural networks, and long short-term memory approaches are implemented for classification. Following this research, it will be evaluated which approaches work best in the classifications of Alzheimer's Disease and Cervical Cancer.

Keywords: Alzheimer, Cervical Cancer**,** Gene Expression, RNA-Seq, Classification, CNN, Deep Neural Network, LSTM, Artificial Intelligence

ÖZET

RNA-SEQ VERİLERİNİN SINIFLANDIRILMASINDA DERİN ÖĞRENME YÖNTEMLERİNİN ENTEGRASYONU

Yayla, Melih

Yüksek Lisans, Elektronik Bilgisayar Mühendisliği

Tez Danışmanı: Dr. Öğr. Üyesi. Bülent HAZNEDAR

Temmuz 2022

103 sayfa

Rahim ağzı kanseri, serviks adı verilen rahmin alt kısmındaki hücreleri etkileyen bir jinekolojik kanser türüdür. Gen mutasyonlarının bir sonucu olarak hücreler anormal hale gelir ve kontrolsüz bölünme kanserin birincil nedenidir. Öte yandan, Alzheimer hastalığı, esas olarak insanlar yaşlandıkça beyin hücresi ölümünün bir sonucu olarak hafıza kaybına ve bunamaya neden olan geri dönüşü olmayan bir nörolojik hastalıktır. Beynin düşünme, öğrenme ve hafızayı kontrol eden kısımları yaralanmış veya tahrip olmuş ve semptomlara neden olmuştur. Rahim ağzı kanseri ve Alzheimer hastalığı her ikisi de genetik bozukluklardır. Birçok genin bilgisi RNA-Seq verilerinde tutulur. Bu yaklaşımı hızlandırmak ve klinisyenlere tanı sürecinde yardımcı olmak için ilişkisiz gen sayısı azaltılarak sınıflandırma algoritmaları kullanılarak metodolojiler oluşturulabilir. Bu önerinin amacı, gerçek örneklerden elde edilen genlerle oluşturulmuş RNA-Seq veri kümelerini kullanarak Rahim Ağzı Kanseri ve Alzheimer Hastalığını incelemek için istatistiksel ve derin öğrenme yaklaşımlarını kullanmaktır. RNA-Seq veri setinin boyutunu küçültmek için iki veri seti için tüm genler arasında %5, %10 ve %30 olacak şekilde gen seçilim yapılır ve gen ekspresyon verileri her bir gen için genlerin önem düzeyine göre oluşturulmuştur. Bu üç senaryoda, seçilen genler kategorizasyon sürecinde eğitilir ve test edilir. Sınıflandırma için Derin Sinir Ağları (DNN), Evrişimli sinir ağları (CNN) ve uzun kısa süreli bellek (LSTM) yaklaşımları uygulanmaktadır. Bu araştırmanın ardından Alzheimer Hastalığı ve Rahim Ağzı Kanseri sınıflandırmalarında hangi yaklaşımların en iyi sonuç verdiği değerlendirilecektir.

Anahtar Kelimeler: Alzheimer, Rahim Ağzı Kanseri, Gen Ekspresyonu, RNA-SEQ, Sınıflandırma, CNN, Derin Sinir Ağları, LSTM, Yapay Zekâ

To My Beloved Family.

ACKNOWLEDGEMENTS

I’d like to thank TÜBİTAK for supporting me in each stage of my work with the 2210-D Master Scholarship Program for Domestic Industry.

I want to thank my supervisor Asst. Prof. Dr. Bülent HAZNEDAR supported and encouraged me during the Master's process. I was a big proud to study with him due to his advice, guidance, and motivation.

A special thanks to my elder brother Faruk YAYLA for his support and assistance throughout the thesis period.

I would like to thank my family and friends for their support. Without that support, I couldn’t have succeeded in completing this project and the thesis.

TABLE OF CONTENTS

Pages

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ANN	Artificial Neural Network
NGS	Next-Generation Sequencing
RNA	Ribonucleic acid
DNA	Deoxyribonucleic Acid
LSTM	Long Short-Term Memory
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
RNA-SEQ	Ribonucleic acid Sequencing

CHAPTER 1

INTRODUCTION

Alzheimer's disease is a kind of dementia that is also a degenerative illness, which unfortunately means it becomes worse over time. Alzheimer's disease is likely to start 20 years or more before symptoms appear, with delicate changes in the brain that go unrecognized by the patient (Villemagne V.L. et al., 2013; Reiman E.M., et al., 2012, Jack C.R. et al, 2009). Individuals only begin to notice symptoms, such as memory loss and language difficulties, after years of brain changes. Nerve cells (neurons) in areas of the brain important for thinking, learning, and memory (cognitive function) have been damaged or destroyed, resulting in symptoms. Alzheimer's disease symptoms are usually present for years(Bateman R.J. et al., 2009; Gordon B.A. et al., 2012). Symptoms tend to worsen with time, interfering with people's ability to go about their daily lives. The person is considered to have dementia owing to Alzheimer's disease, or Alzheimer's dementia, at this moment(Braak H. et al., 2011).

The prevalence of Alzheimer's disease is estimated to be 5% in those aged 65 and up, but it's startlingly higher (about 30%) among people aged 85 and above in industrialized nations. As worldwide life expectancy rises, the number of Alzheimer's sufferers is predicted to rise dramatically. According to a recent study, Alzheimer's disease is the 6th biggest reason of death in the US. Approximately 16 million Americans provide unpaid care to 6 million Americans suffering from Alzheimer's disease, bringing the total to 18.5 billion hours worth $234 billion (Friedman E.M. et al., 2015; Spillman B. et al., 2011; Walmart, 2018; McDonald’s, 2017; Rabarison K.M. et al., 2018).

Other areas of the brain are injured or destroyed as the disease progresses. Activities that used to be central to a person's identity, such as organizing family gatherings or engaging in sports, may no longer be possible. Neurons in regions of the brain that allow a person to walk and swallow are eventually impaired. Alzheimer's disease patients are bedridden and require 24-hour care. Alzheimer's disease is deadly in the long run (Gaugler J. et al., 2019).

In addition to disorders such as Alzheimer's, cystic fibrosis, hemophilia, Mediterranean anemia, and several forms of cancer (e.g., breast, colon, ovary) that can now be identified with DNA-RNA, test methods for the diagnosis of the estimated

4000 hereditary diseases are required. As a result, among the mapped genes in the genome, it is important to discover a function for gene sequences of unknown function. Only 2% of the genes in DNA can be converted into protein after they are translated into RNA. It is unable to convert the remaining 98 percent of the DNA to protein, i.e., it is unable to code. Gene sequences with unclear functions are also found in this non-coding section. The more illness-causing genes are discovered and understood, the easier it will be to produce medications tailored to the patient and hence to the condition and to employ them for therapy. Cluster analysis' major goal is to locate groups of people that have similar expression patterns, which might lead to the identification of novel cancer subtypes (Datta and Nettleton, 2014).

According to the World Cancer Report published by the World Health Organization in 2008, cervical cancer is the second type of cancer seen in women worldwide (Boyle P. and Levin B., 2008). While approximately 400,000-500,000 new cervical cancers are detected in the world every year, 190,000 of them die and 78% of deaths occur in developing countries. These numbers show how important screening programs for cervical cancer are in reducing the incidence of this cancer and the death rate from this cancer (Tuncer M., 2009). According to the World Health Organization 2009 report, the annual number of new cases worldwide is 493,243 and the number of deaths from cervical cancer is 273,505 (Sönmez Y., 2012). According to the same report, it is estimated that the number of new cases will be 756,043 in 2025 and 438,884 of these patients will die (Kurt A. et al. 2013).

On the other hand, Human Papilloma Virus (HPV) was found to be responsible for 99.7% of cervical cancer (Camcıoğlu Y., 2008). HPV infection rates range from 32.9% to 69.0% in sexually active adolescents (Welling K., 2001). According to the reports from the Center for Disease Control and Prevention (CDC), 6.5 million people are infected with HPV every year in the United States (USA), and a total of 20 million people are infected with HPV. The International Agency for Research on Cancer (AIRC) expects a 40% increase in cervical cancer globally by 2020. While an increase of 50-55% is expected in countries with low socioeconomic status such as Africa, Latin America, and Asia, an increase of 6% is expected in Europe and 23% in North America (Camcıoğlu Y., 2008). The transmission of HPV infection can occur by sexual contact or by contact with the genital area without penetration during sexual intercourse. In addition, the frequent asymptomatic course of the virus after infection reduces the

chance of early diagnosis and treatment of the disease, causing cervical cancer to occur in the later stages of the disease (Boyle P. and Levin B., 2008). The most important feature of cervical cancer is that it is preventable and the prognosis of the disease is good when it is diagnosed early (Barut A., 2008). Cervical cancer screening is traditionally performed with a regular pap smear test, colposcopy/biopsy, and HPV Deoxyribonucleic Acid (DNA) test (Camcıoğlu Y., 2008).

Many areas such as deep learning, machine learning, and data mining have been studied to improve the classification accuracy of Cervical cancer and Alzheimer's, minimize the classification errors of false positive and false negative records, and identify the most relevant risk factors for both diseases. Some examples of these studies are briefly summarized below.

Rayavarapu and Krishna used two popular machine learning (ML) techniques, a Voting classifier, and a Deep neural network classifier. The dataset they used in their study is the cervical cancer risk factors dataset in the UCI repository. In their study, they conducted on the target variables of Cytology and Biopsy, two of the four target variables in the data set. When the results were compared in terms of accuracy, the Voting Classifier achieved the highest accuracy (97–99) for cytology and biopsy target variables when compared to the DNN (Deep Neural Network) classifier (Rayavarapu K. and Krishna K. K. V., 2018).

Hyeon et al utilized a pre-trained convolutional neural network (CNN) as a feature extractor and several machine learning classifiers to classify cell images as normal and abnormal. Logistic Regression, Random Forest, AdaBoost, and Support Vector Machine (SVM), which are machine learning classifiers, were used in the study. Studies were carried out in MATLAB software. At the end of the studies, the support vector machine achieved the highest performance with an F1 score of 78% (Hyeon J. et al., 2017).

Six different machine learning and data mining algorithms, including k-nearest neighbors (k-NN), decision tree (DT), rule induction, Naive Bayes, generalized linear model (GLM), and deep learning algorithm, are applied to the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset in Shahbaz et al. study to classify the five different stages of Alzheimer's disease and to identify the most distinguishing attribute for each stage of AD among the ADNI dataset. The study's findings indicated that the

GLM can accurately diagnose the phases of Alzheimer's disease with an accuracy of 88.24% on the test dataset (Shahbaz M. et al., 2019)

Genes are microscopic portions of DNA that store all of a cell's memory. This DNA stores all of the genetic information needed to make every protein the cell need. Each gene has a specific set of instructions that code for a certain protein. Gene expression is the process of a gene's creation with all of the required information. Gene expression, in a nutshell, depicts the status of a gene's activation during the production of a protein. Microarray is a laboratory instrument for detecting the expression of a large number of genes at the same time. mRNA molecules from two separate persons are acquired for microarray analysis. The first reference may be a healthy person, whereas the second could be someone who has a condition such as cancer. The information acquired by microarrays can be utilized to diagnose and classify human cancer (Perez-Diez A. et al.,2007). Because of a few key benefits, microarray technology gave way to RNA-Seq technology, and RNA-Seq technology became the dominant principle in gene-expression research (Ritchie M.E. et al., 2015). With so many genes, RNA-Seq data have high dimensionality. The majority of genes are irrelevant during cancer and Alzheimer's diagnosis. For example, there are about 25.000 coding genes in the human genome, with 291 of them linked to cancer (Futreal P.A. et al., 2004). This study shows that the number of genes used in cancer categorization and diagnosis may be reduced. In RNA-Seq datasets, genes are utilized to categorize or diagnose cancer and Alzheimer's disease. The performance of your classification or diagnostic system is heavily influenced by feature selection.

With microarray data, several types of feature selection approaches and classification algorithms have been investigated for cancer detection. Recent research has employed Bayesian, Support Vector Machine, Decision Tree, Artificial Neural Networks, Random Forest, and Bayesian classification algorithms. A. Statnikov, for example, produced a cancer diagnosis using microarray data, and the greatest classification performance for leukemia data was reached by utilizing SVM with a 97.5 percent accuracy (Statnikov, A. et al., 2005). Mahendran N. et al used microarray data and the Improved Deep Belief Network (IDBN) and Bayesian Optimization method to improve the classification of Alzheimer's disease (Mahendran N. et al, 2011).

Deep Neural networks, convolutional neural networks, and long-term short memory approaches are evaluated in this study to discover the best classification model for Cervical cancer and Alzheimer's disease. Algorithms are used for RNA-Seq datasets from Alzheimer's disease and cervical cancer. Gene selection scenarios of 5%, 10%, and 30% of gene selection will be submitted to the system, which will be analyzed and tested to see how genes are correlated to conditions. Adadelta, Adam, Adamax, and RMSProp are the four optimizers that will enhance the classifiers. After the investigation, the acquired findings of algorithm comparisons are shown. The findings will provide researchers with a better grasp of how to apply feature selection and classification algorithms to RNA-Seq data.

The thesis is organized as follows: Chapter 2 covers the biological background of cervical cancer and Alzheimer's disease, gene expression, RNA-seq technologies, data structure, applied pre-processing methods such as data normalization and transformation, as well as related works on classification algorithms and feature selection. In Chapter 3, all classification techniques are first explained, then methods of classification evaluation are displayed, then tools that are utilized in this study, training, and testing methods are given in detail, and finally, classifiers network structures are shown. In Chapter 4, the experimental results of two independent RNA-Seq datasets are applied and reported in detail using optimizers and classification algorithms. Finally, Chapter 5 concludes with the last thoughts on the subject and recommendations for further research.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

2.1. Cervical Cancer

Cervical cancer is the second most frequent cancer in women aged 15 to 44, after breast cancer. In comparison to other cancers, its overall incidence was 14.8 per 100,000 women, with an age-specific incidence of 8.3 per 100,000 women. Cervical cancer is the fifth leading cause of mortality among women aged 15 to 44, and the fourth leading cause of death among women overall (ICO HPV Information Centre, 2016).

Cervical cancer, which is one of the gynecological diseases for which early detection is simple and inexpensive, has a significant death rate if not detected and treated early. Cervical cancer, the second most common type of gynecological cancer in the world after breast cancer, is a public health issue that continues to be a global issue due to its high incidence, high mortality and morbidity rates, and the financial burden it places on health systems (Günaydın, 2013; Işık et al., 2016).

The most important known risk factor for Cervical Cancer is HPV (Human Papilloma Virus) infection. Other risk factors are the situations in which the immune system is suppressed such as smoking, HIV (AIDS) infection, experiencing more than three pregnancies, exposure to diethylstilbesterol (DES), a synthetic estrogen in intrauterine life, having a family history of cervical cancer, being overweight, using oral contraceptives for a long time, poor nutrition from fruits and vegetables in the diet, low socioeconomic level, Sexually transmitted infections (Herpes, chlamydia, gonorrhea, syphilis) (American Cancer Society, 2017; Mishra, G.A. et al, 2016).

HPV infection is divided into three stages: latent, subclinical, and clinical. The illness has no cytological or morphological manifestations during the latent stage, and HPV DNA can only be detected with ultrasensitive PCR methods. The clinical period is when there are obvious lesions and symptoms, such as genital condyloma or invasive malignancy. Cervical cancer is characterized by painless bleeding and a broth-colored vaginal discharge. Abnormal vaginal bleeding, postcoital or postmenopausal bleeding, spotting between periods, prolonged and heavier menstrual bleeding, bleeding after pelvic examination, dyspareunia, foul-smelling vaginal discharge in advanced stages, low back and groin pain, single or cauliflower appearance on genital and anal mucosa

Numerous painless lesions, anemia, weight loss, difficulty urinating or leg edema can be seen (Aydoğdu and Özsoy, 2018).

2.2 Alzheimer

Alzheimer's disease, which is the most common type of dementia, is defined as a progressive neurodegenerative disease that causes intellectual decline, various neuropsychiatric behavioral disorders, and disturbances in daily life activities *(Selekler, 2003).* Alzheimer's disease is mostly caused by aging. Family history of dementia, gender, low education level, genetic factors, down syndrome, history of major depression, vascular events, plasma homocysteine level, hypothyroidism, hypercholesterolemia, myocardial infarction, atherosclerotic carotid disease, and hypertension are the other risk factors. (Selekler K., 2003; Gürvit H. İ., 2010).

Reisberg, the developer of the Global Deterioration Scale, which is one of the two systems widely used to characterize the stages of Alzheimer's disease and cannot be used in other dementias due to the specificity of the disease, said that this process in Alzheimer's patients is the complete reversal of the progressive individuation-independence developmental process of humans in infancy, late childhood and adolescence claims and calls this progressive destruction retro genesis (Gürvit H. İ., 2010). Alzheimer's patients at the mild-dementia stage, according to Gürvit (2010), resemble school-aged children aged 7 to 12, who have achieved a certain amount of independence in and out of the house but require supervision in complex activities that involve social interactions or judgment. A middle-stage Alzheimer's patient is like a preschool child between the ages of 2 and 6 who requires supervision in basic daily life activities such as dressing, eating, and washing; a severe-stage Alzheimer's patient is like a 0-2-year-old baby who is completely reliant on their parents (caregivers) for 24 hours to maintain their life (Gürvit H. İ., 2010). The Clinical Dementia Rating Scale (Clinical Dementia Staging) is the other second staging scale utilized. In this scale, Alzheimer's disease is categorized into three stages: mild, moderate, and severe, respectively (Selekler K., 2003).

According to Gürvit (2010), mental abnormalities that can be classified as Alzheimer's disease are severe enough to produce considerable impairments in daily life activities. Gürvit divided the substantial deteriorations into three categories which are cognitive impairments, behavioral degradation, and functional impairments.

Other problems that should be examined and analyzed include disturbances in motor, autonomous, and sleep behaviors. Alzheimer's disease is distinguished from other dementia disorders by its time and spatial focus.

2.3. Gene Expression

Proteins, which are generated utilizing the information encoded in genes as DNA, control the biological operations of a living creature. This genetic information is translated into proteins after being transcribed into RNA. Different cell types and proteins produced within cells are specified by the transcription of specific genes into a collection of RNA molecules (called the transcriptome), and cellular activities are governed by created proteins. As a result, the expression level of associated genes and environmental factors control the number of RNA molecules and proteins produced. These RNA molecules, as well as gene expression levels, are critical for understanding cellular functions, cell growth, and illness (Kukurba K.R. and Montgomery S.B., 2015). Microarray technology was employed in the initial gene expression research. However, with the advent of next-generation sequencing (NGS) technology, RNA-sequencing has become popular and commonly favored for gene-expression research in recent years. There are three types of commercial NGS platforms (Zararsız G.,2015):

1. Second-generation platforms like Illumina HiSeq, Roche 454, and ABI/SOLID,

2. Third-generation platforms like Ion Torrent, Pacific Bio, and Complete Genomics, and

3. Fourth-generation platforms like Oxford Nanopore. In terms of computing speed, accuracy, sequencing depth, output data size, and other factors, each platform has its own set of benefits and drawbacks.

Furthermore, the sequencing platform used might be a key determinant for determining approaches for downstream analysis and interpretation. As a result, an effective sequencing platform should be established by concurrently taking into account experimental aims, benefits, and drawbacks. Proteins, which are generated utilizing the information encoded in genes as DNA, control the biological operations of a living creature. This genetic information is translated into proteins after being transcribed into RNA. Different cell types and proteins produced within cells are

specified by the transcription of specific genes into a collection of RNA molecules (called the transcriptome), and cellular activities are governed by created proteins. As a result, the expression level of associated genes and environmental factors control the number of RNA molecules and proteins produced. These RNA molecules, as well as gene expression levels, are critical for understanding cellular functions, cell growth, and illness (Kukurba K.R. and Montgomery S.B., 2015). Microarray technology was employed in the initial gene expression research. However, with the advent of next-generation sequencing (NGS) technology, RNA-sequencing has become popular and commonly favored for gene-expression research in recent years. There are three types of commercial NGS platforms (Zararsız G.,2015):

1. Second-generation platforms like Illumina HiSeq, Roche 454, and ABI/SOLID,
2. Third-generation platforms like Ion Torrent, Pacific Bio, and Complete Genomics, and
3. Fourth-generation platforms like Oxford Nanopore. In terms of computing speed, accuracy, sequencing depth, output data size, and other factors, each platform has its own set of benefits and drawbacks.

Furthermore, the sequencing platform used might be a key determinant for determining approaches for downstream analysis and interpretation. As a result, an effective sequencing platform should be established by concurrently taking into account experimental aims, benefits, and drawbacks.

2.4. Rna-Seq

Using one of the NGS systems, RNA sequencing is utilized to gather gene expression data from transcriptomes. The Illumina HiSeq technology has recently become a popular and widely used RNA sequencing tool. Although the RNA sequencing procedure may alter based on the sequencing platform, the basic concept remains the same. RNA sequencing, on the other hand, has a more extensive approach that starts with experimental design and ends with feature counting. To produce impartial and reliable data, the right experimental design should be selected in the initial phase of the RNA sequencing method. The core concepts of a successful experimental design are randomization, replication, and blocking. In RNA sequencing, there are two main types of replications: biological and technological replications. A

technical duplicate is a set of two measurements made twice from the same topic. A biological replication, on the other hand, is a pair of measurements collected from two distinct samples. Technical replicates are often favored to enhance sequencing depth and better discover differentially expressed genes, even though both types of replicates can boost statistical power. Liu et al. (2014) showed that regardless of sequencing depth, adding additional biological replicates improves statistical power (Liu Y. et al., 2014). During sample preparation, randomization and blocking are used to assign samples to each block and group at random. As a result, defining an effective library preparation process for detecting RNA molecules of interest is critical. Ribosomal RNA molecules, for example, make up 95 percent of the cell's total transcriptome. If these molecules are not eliminated before library formation (i.e. the region of interest to which sequenced samples will be mapped), ribosomal RNA will be linked with a significant portion of mapped reads. As a result, because the sequencing reads of less abundant RNAs are relatively small, they will not be appropriately recognized. As a result, unnecessary RNA molecules are removed throughout the in-sample and library preparation phases, and interesting gene areas are identified. Isolated RNA molecules are sequenced using the NGS platform in the quality assessment and alignment stage, and millions of short sequence reads are generated in a single run, along with a quality score for each read. Raw sequencing reads are then preprocessed, low-quality reads filtered, and filtered short reads matched to the reference genome. In the feature counting stage, mapped reads are tallied for each gene region (Göksülük D., 2019). Finally, a typical RNA sequencing experiment comprises the following procedures in order:

1. RNA molecules are extracted from the transcriptome and broken into 200-base units.

2. Complementary DNA is created from fragmented RNA molecules (cDNA).

3. The sequenced cDNA segments are aligned to the reference genome.

4. Counts of mapped reads are obtained.

2.5. Data Structure of Rna-Seq

Microarrays and RNA sequencing are examples of high-throughput technologies that produce a gene expression data matrix X with each row representing gene areas and each column representing samples. Microarray data is returned on a

continuous scale, whereas RNA sequencing data is returned on a discrete scale. Assume that three healthy and three sick individuals' mRNA molecules are sequenced and aligned to 500 distinct reference gene regions. Table 2.1 illustrates an example of possible RNA sequencing results. The total mapped read counts of the jth sample to the ith gene are represented by each cell in the data matrix (x_{ij}).

Table 2.1: An example of RNA sequencing data

Features	Healthy			Diseased			Total
	H1	H2	H3	D1	D2	D3	
Gene 1	304	3	14678	92	177	1	97865
Gene 2	0	0	438500	7600	47387	62	9472601
Gene 3	14	26	972	48	714	184	74318
…	…	…	…	…	…	…	…
Gene 500	0	1	72	14390	28400	187	162840
Total	6792	1376	12765432	151100	338405	9320	89366412

These numbers are proportional to the levels of gene expression. As a result, the number of mapped read counts rises in tandem with the activity of linked genes. However, mapped read counts are affected by a variety of factors, including sequencing depth, gene length, sequence quality, and so on. As a result, raw counts cannot be utilized as a direct measure of gene expression level unless they are preprocessed for downstream analysis like differential expression, categorization, clustering, and so on. Raw counts, for example, should be adjusted to eliminate sample variances like sequencing depth and compare all samples on the same scale. Raw counts might be converted in a similar way to be utilized in clustering and classification tasks for continuous data.

2.6. Normalization of Rna-Seq Data

Depending on the sequencing depth and gene length, the total amount of mapped reads for each sample and gene may be extremely varied. Depending on the sequencing depth, the mapped read counts to the same feature of two distinct participants might change dramatically. Similarly, depending on the length of the gene, the mapped read counts to various genes of the same subject may change. As a result, in RNA-Seq research, two values $x_i = \sum_{j=1}^{n} x_{ij}$ and $x_j = \sum_{i=1}^{p} x_{ij}$ are dependent on experimental design, resulting in technical biases in downstream analysis. As a result,

raw read counts should be normalized before proceeding with further studies, such as differential expression analysis for discovering relevant genes, sample categorization, and/or grouping using machine learning methods. As a consequence, it is self-evident that size factors should be estimated to normalize raw numbers. Rather than utilizing estimations s_{ij} for each cell, all normalization approaches addressed in this thesis employ a single-size factor estimation s_j for each subject, and mapped read counts for each gene are globally normalized using that subject's size factor sj. Although multiple size factors for each feature might be estimated for the j-th sample, we normally assume that the size factor estimate for each sample is constant for all features to reduce model complexity.

Previous microarray studies have demonstrated that normalization is an important step in differential expression analysis for reducing false-positive findings and obtaining reliable estimations (Robinson M.D. and Oshlack A., 2010). Although RNA-Seq yields less noisy data than microarrays, normalization is still an important consideration since it allows researchers to compare gene expression levels between two samples with different sequencing depths. The total number of reads in a sample is referred to as sequencing depth or library size x_j. We anticipate the ratio of expected counts $E(X_{ij})/E(X_{ij'})$ for subjects j and j' to be equal to the ratio of size factors $s_j/s_{j'}$ for the same subjects if the ith gene is not differentially expressed. As a consequence, it is self-evident that size factors should be estimated to normalize raw numbers. Rather than utilizing estimations sij for each cell, all normalization approaches addressed in this thesis employ a single size factor estimation sj for each subject, and mapped read counts for each gene are globally normalized using that subject's size factor s_j. Although multiple size factors for each feature might be estimated for the j-th sample, we normally assume that the size factor estimate for each sample is constant for all features to reduce model complexity.

The RNA-Seq experiment uses a different data production pipeline than microarrays. As a result, the microarray normalization approaches –also known as size factor estimation– are not directly relevant to RNASeq data. Several normalization strategies for RNA-Seq data have been developed in recent publications (Robinson M.D. and Smyth G.K., 2008; Oshlack A. and Wakefield M.J., 2009; Trapnell C. et al., 2012). Total count, median ratio, and upper quartile normalization methods will be

discussed and used. References (Cloonan N. et al. 2008; Parikh A. et al., 2010; Robinson M.D. et al., 2011; Sultan M. et al. 2008) provide information on other normalization approaches like the trimmed mean of M-values, quantile, reads per kilobase per million mapped reads (RPKM), CuffDiff, PoissonSeq, and so on. The majority of suggested genomics data techniques are utilized to find differentially expressed genes. However, when computing the size factors of training and test samples, additional steps are necessary for classification and clustering tasks (Mortazavi A. et al., 2008). Because we should estimate the size factor s of a test sample using training set characteristics, the size factor estimation processes for a test sample $x = x_1^*, x_2^*, \ldots, x_p^*$ are not easy. Finally, x_{ij}/s_j for training set samples and x^*/s^* for the test, samples are used to generate normalized read counts with estimating size factor as s^* of a test sample by using training set parameters.

2.6.1. Deseq Median Ratio Normalization

The median-of-ratios approach suggested by Love et al. is a reliable method for total count normalization for size factor estimate (Love M.I. et al. 2014). $s_j = m_j / \sum_{j=1}^{n} m_j$ is used to calculate the size factors, where m_j is defined as

$$m_j = \underset{i:\, G_i \neq 0}{\text{median}} \left\{ \frac{x_{ij}}{G_i} \right\}, \quad G_i = \left(\prod_{j'=1}^{n} x_{ij'} \right)^{1/n} \tag{2.1}$$

Here, Gi represents the geometric mean of reading counts for the i-th feature, and m_j represents the geometric mean of features with a nonzero geometric mean.

$$s^* = \frac{m^*}{\sum_{j=1}^{n} m_j}, \; m^* = \underset{i:\, G_i \neq 0}{\text{median}} \left\{ \frac{x_{ij}}{G_i} \right\} \tag{2.2}$$

is used to compute the size factor of a test sample x^* where m_j and G_i are calculated from the training set using the above equation. The denominator G_i is a sample that is used to compare each sample (Evans C. et al. 2017)

2.7. Transformation of Rna-Seq Data

The majority of suggested methods for RNA-Seq differential expression analysis are based on discrete distributions like Poisson and negative binomial, and raw counts are employed directly without processing (Love M.I. et al. 2014; Robinson M.D. et al., 2010; Hardcastle T.J. and Kelly K.A., 2010). For other uses, such as clustering, classification, and graphical representations, however, working with modified data may be a better option for a variety of reasons. For starters, the variation

of reading counts for a gene is significantly reliant on the anticipated count for that gene, which leads to the heteroskedasticity problem in RNA-Seq data. Because a large part of these techniques works well with normally distributed homoscedastic data, this issue causes erroneous clustering and classification results. Second, the mapped read counts are slanted to the right and distributed. Finally, from the point of view of mathematical theory, discrete distributions are less common than normal distributions. Consequently, the performance and use of discrete distributions in RNA-Seq studies are often limited (Law C.W. et al. 2014). The purpose of the census change is to obtain approximate data of normal and equal variation. Consequently, the transformations allow researchers to apply statistical approaches developed for RNA-Seq studies, especially for normally distributed microdata, such as hierarchical grouping, linear discriminant analysis, and thermal maps. $X \rightarrow Z$ can be the simplest transformation:

$$z_{ij} = \log_2(x_{ij} + c),\ Z_{ij} \sim Normal(\mu_{ij}, \sigma_{ij}^2) \qquad (2.3)$$

where c is a small quantity that prevents the transformation from having a logarithm of zero is a constant of. C also plays a role in distribution rights. The transformation (equation) is also known as shifted logarithmic transformation because of this. The logarithmic transformation transforms a right-skewed distribution into a nearly symmetric one. The fact that extremely high read counts have smaller weights and very tiny ones have excessive weights in translated space is a significant disadvantage of this transformation. As a result, increasing false-positive rates for finding differentially expressed genes on a logarithmic scale are more likely to be seen (Law C.W. et al. 2014). For RNA-Seq data, more robust and complex transformation algorithms have recently been presented. The most frequent ones, such as variance stabilizing (VST) and power transformations, will be explained and covered in this section.

2.7.1. Variance Stabilizing Transformation

One of the proposed strategies for making estimated variances independent of the means is the variance stabilizing transformation. When data has a lot of overdispersions, the variance of log_2 scaled counts are usually larger than the mean, and this reliance reduces as the mean rises. The variance stabilizing transformation eliminates a large portion of the mean-variance dependency, resulting in converted values with a nearly constant mean-variance trend.

Consider the mapped read counts of a random variable X with the mean-variance relationship $\phi^2=\mu+\phi\mu^2$ and ϕ as an overdispersion parameter. Anders and Huber (2010) used the following parametrization to study the link between the mean and the overdispersion parameter(Anders S. and Huber W., 2010):

$$\phi = \phi_0 + \eta/\mu \quad (2.4)$$

where $\phi_0$denotes asymptotic overdispersion and is an extra-Poisson parameter. The following formula may be used to compute variance as a function of the mean:

$$v(\mu) = \sigma^2 = \mu + \mu^2\phi_0 + \mu\eta \quad (2.5)$$

A variance stabilizing transformation $\tau(.)$generates a transformed value z as a consequence.

$$\tau(x) = z = \int^z \frac{1}{\sqrt{v(\mu)}} d\mu \quad (2.6)$$

2.7.2. Power Transformation

For RNA-Seq classification, Witten (2011) introduced the Poisson model. The mean and variance of reading counts are assumed to be equal in this model(Witten D.M., 2011). This assumption, however, is broken since the variation of reading counts is frequently larger than the mean. As a result, to bring variation around the mean, a power transformation is done to count data. When there is mild to moderate data dispersion, power transformation works effectively. Power transformation, unlike vst transformation, does not consider the size. As a result, before continuing with the classification, altered data should be normalized.

2.8. Literature Review

Developing computer methods for gaining insights and extracting features from data has been a growing challenge in recent years. Many advancements have been made through data mining, but artificial intelligence research is becoming more popular as a result of the massive increase in computational power (Kumar D. and Sharma D., 2019). Due to the unusually large quantity of gene expression data compared to a typically limited number of existing samples in the same scope, analysis based on microarray technology encounters major obstacles in terms of processing these analyses in the bioinformatics field (Chen W. et al., 2009).

Chen et al introduced the categorization of gene expression data using artificial neural network ensembles based on sample filtering in research published in 2009. The suggested technique was validated utilizing leukemia datasets in simulation tests, and the results were compared to those of a single neural network, bagging neural network ensembles, and a support vector machine. Their findings revealed that their approach is both steady and precise (Chen W. et al., 2009).

In their study about Gene Expression published in 2019, Kumar D. et al. showed that Deep Learning, a more recent branch of machine learning, is required for various models and theoretical foundations, as well as motivations for why we need deep learning in the context of evolving big data, particularly in the area of gene expression level classification. They also offered a review of several DL models, their merits and shortcomings, and their computing capacity for gene expression modeling in the same article (Kumar D. and Sharma D., 2019).

In their paper named "Pattern Identification and Classification in Gene Expression Data Using an Auto associative Neural Network Model" published in 2002, Bicciato S. et al. stated that using microarray technology to analyze gene expression offers up tremendous prospects to better understand biological systems and speed up the discovery of target genes and pathways for medicinal chemistry and psychosocial interventions. Parallel monitoring of hundreds of genes' expression patterns, as indicated in this paper, appears to be particularly promising for a better knowledge of cancer biology and the development of molecular fingerprints that support histological categorization systems of neoplastic specimens. According to Bicciato S. et al., examination of the neural network's internal structure allows for the discovery of unique phenotypic markers and the inference of distinctive correlations between genes and physiological states, while neural network outputs are for unique cases. They also highlighted that it allows for the designation of several classes in the same research, such as distinct clinical states or tissue samples (Bicciato S. et al., 2003)

Non-Coding (NC) RNA, according to another study by Amin N. et al., plays an important function in biological processes and has been linked to illnesses like cancer in 2019. The categorization of noncoding RNAs was utilized in this work to better understand the mechanisms behind illnesses and to develop effective therapies.

Deep learning was performed to identify and classify ncRNAs, with encouraging findings in this study (Amin N. et al., 2019).

In research published in 2014, Tan K.M et al. stated that next-generation sequencing technologies provide for a detailed snapshot of RNA transcripts included in a tissue sample at a reasonable cost. The generated readings are frequently categorized by gene, exon, or another region of interest in their research; thus, data generally refer to read counts for tens of thousands of features in dozens or hundreds of observations. They also mentioned that using this data to generate a classifier and assign an observation to one of many predetermined classes is an intriguing technique. Statistical challenges are also prevalent in RNA-SEQ classification due to the large dimensionality of the data, and many existing classification algorithms cannot be applied directly because there are many more characteristics than observations, as indicated in the gene expression research. Traditional classification methods, such as logistic regression, linear discriminant analysis, principal component analysis, partial least squares, and the use of a support vector machine in a high-dimensional environment, as well as its modifications, were used in two RNA sequencing datasets, according to their research (Tan K.M et al., 2014).

Zararsız G. et al. stated in their 2017 paper that RNA-Seq technology is a new and efficient approach for characterizing and quantifying transcriptomes that harnesses the capabilities of next-generation sequencing technology. In the same study, they also stressed the need of identifying a limited group of genes that may be utilized to develop disease diagnostic classifiers. The concern that Microarray-based classifiers cannot be directly applied to RNA-Seq data due to their discrete form is discussed in this paper. Another issue raised is overdispersion, which necessitates careful modeling of the RNA-Seq data's mean and variance relationships (Zararsız G. et al, 2017).

According to Göksülük D. et al., RNA-SEQ technology has become the favored approach in recent years and provides less noisy data, making it preferable to microarray technology for gene expression-based categorization and differential expression analysis. Although there are many suggested algorithms for microarray data, the number of algorithms and tools available for the categorization of RNA sequencing data is restricted, according to the same study. As a result, they created MLSeq, a machine learning interface that brings together not just frequently used

classification algorithms but also unique techniques and makes them available for use in RNA sequencing data categorization. MLSeq package is freely available.

Most suggested statistical approaches for the classification of gene expression data either rely on a continuous scale (eg, microarray data) or need an assumption of normal distribution, according to another study on the simulation of RNA-Seq Classification by Zararsiz G. Goksuluk D. et al. They found that the aforementioned methodologies could be used directly to RNASeq data with proper adjustments, ensuring that neither data structure nor distribution assumptions were violated. They discovered that one method is to create count-based classifiers like Poisson linear discriminant analysis and negative binomial linear discriminant analysis, and another is to approximate data to microarrays and use microarray-based classifiers. They compared PLDA with and without power conversion, NBLDA, single SVM, bagging SVM, classification and regression trees, and random forests in their work. They also looked at the impact of several factors on model performance, such as overdispersion, sample size, number of genes, number of classes, differential expression rate, and transformation technique, and reported their findings. According to these findings, increasing the sample size, differential expression ratio, and reducing the distribution parameter and the number of groups improved classification accuracy. They also mentioned that, like differential expression studies, stratification of RNA-Seq data demands prudence when dealing with data that is over-dispersed. They found in the same study that power transform PLDA as a Number-based classifier, as well as vst or rlog-converted Random Forest and SVM classifiers as microarray-based classifiers, might be suitable alternatives for classification (Zararsız G. et al, 2017).

Therefore, developing a method by integrating current and innovative classification methods such as artificial intelligence and deep learning instead of classical classification methods in the mentioned RNA sequencing dataset and making them usable for the classification of RNA sequencing data is among the most important original goals of this thesis study. On the other hand, statistical learning problems in many fields involve sequential data. Traditional machine learning assumes that data points are independently and uniformly distributed (Yadavendra and Chand S., 2020), but in sequencing data, a data item depends on what comes before or after it. Recurrent neural networks (RNN) are a well-known method in sequence models Hochreiter S. and Schmidhuber J.,1997) and will be among the methods to be used in this study.

Convolutional neural networks (CNN), another method that will be included in this study, were used for the classification of sequencing datasets in another study and it was seen that it gave superior performance compared to many modern approaches (Cheng L., et al 2022).

Chakraborty R. et al. mentioned in their research in 2019, that CNN and LSTM indicated their capacity to include feature extraction and natural language processing, respectively. As a result, in their research, they attempted to harness their capacity to understand RNAs' language, i.e., anticipate the microRNA sequence from the mRNA sequence. The goal of this research is to extract characteristics from mRNA in sequence form and predict miRNA using an LSTM network. While revalidating experimentally confirmed data, their model was able to predict an average of 72 percent of miRNAs for given mRNA. In comparison to other predicted methods, it was also able to exhibit the largest positive expression fold change of projected targets in microarray data collected utilizing anti-25 miRNAs (Chakraborty R. and Hasija Y., 2020).

CHAPTER 3

METHODS AND MATERIALS

3.1. Classification Algorithms

3.1.1. Artificial Neural Network

Artificial neural networks are primarily concerned with mathematical modeling of the nervous system and living things' decision mechanisms. Its goal is to use artificial neural networks that can be learned, self-organized, and flexible to artificially imitate the learning structure of the living brain. This method is supposed to provide robots with a human-like learning and decision-making process (Alpaydın E., 2011)

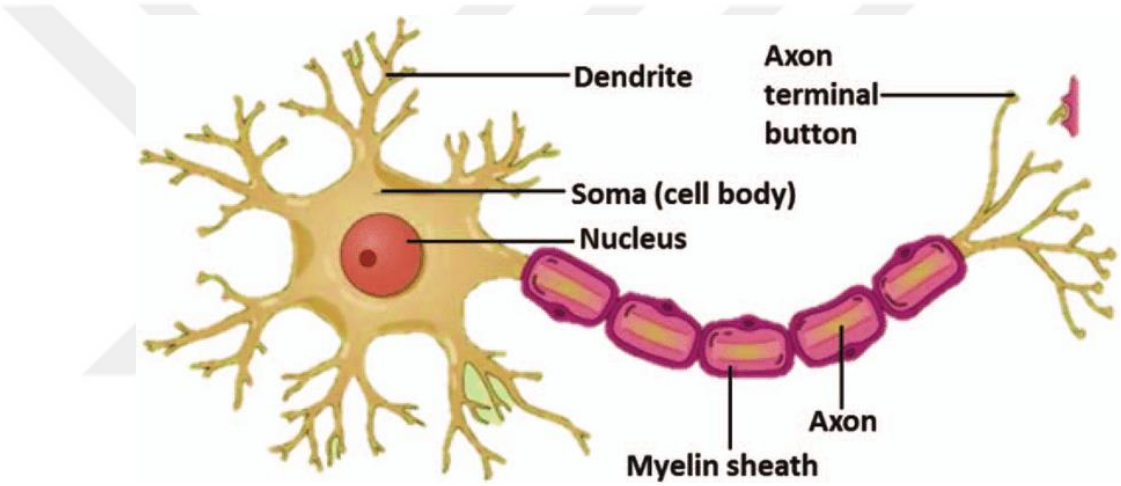

Figure 3.1: Human Neural Cell Structure (Rai P. et al, 2013)

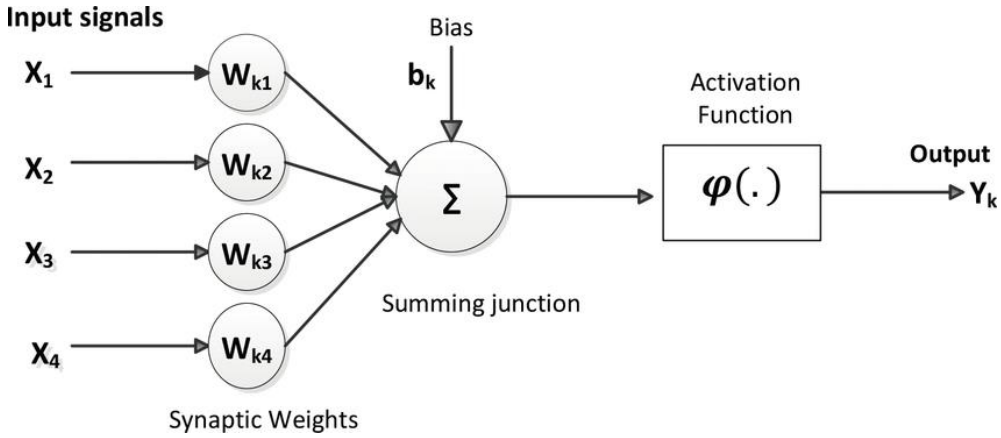

Figure 3.2: Computational Model of a Neuron (Guesmi, L. et al, 2018)

Dendrites serve as the system's input, while axons serve as the system's output. Inputs and outputs are connected via synapses. The electrical signal is generated regularly by the core. Myelin is an insulating substance that influences the pace of spreading (Kızrak M. A. and Bolat B., 2018). Chemical carriers transmit the signal carried by the axon to the synapses. For cell stimulation, the voltage over a particular threshold value must be provided; otherwise, nerve conduction will be blocked. The perceptron, also known as the smallest component of an artificial neural network, is represented by the equation below and a linear function (Rosenblatt F., 1957)

$$y = W.x + b \tag{3.1}$$

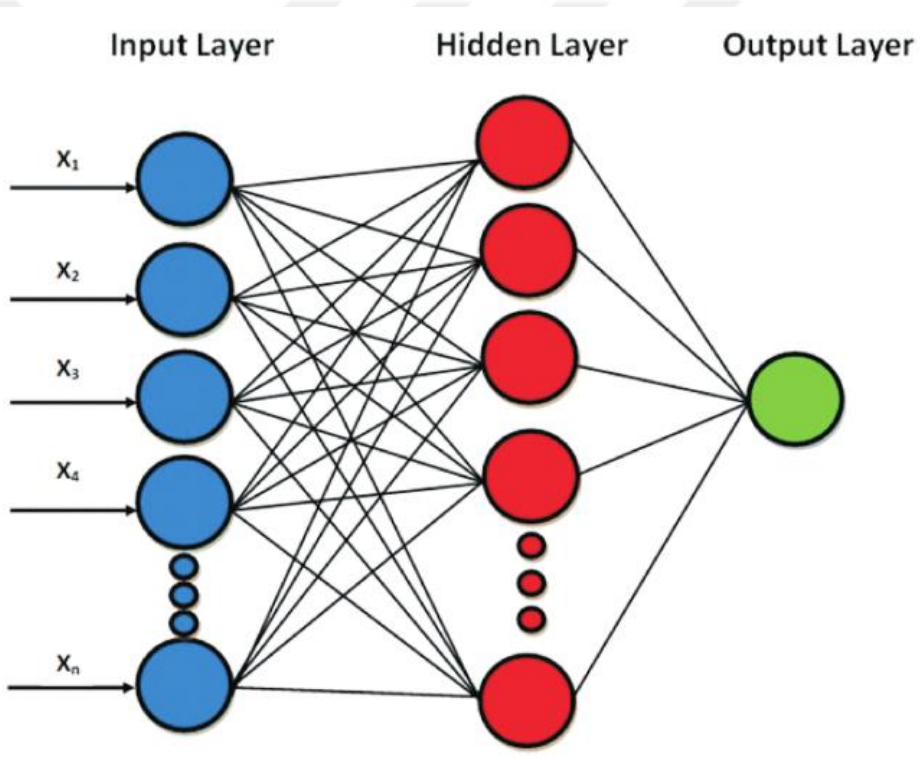

Figure 3.3: Structure of the Artificial Neural Network Model (Khademi F. and Jamal S. M., 2016)

Each layer affects the model, and each neuron in each layer affects the layer and, as a result, the model. In a traditional neural network, neurons in each layer have no connections with one another, and data is just sent from one layer to the next or output. Neurons in two consecutive layers have different activation values that impact each other. The number of neurons in the layers has an indirect impact on the system's performance. One of the most significant topics to consider when building an artificial

neural network model is how the starting values of the weight vectors should be established. While the weights can be allocated at random at first, the weights of a previously trained model can alternatively be used as the starting weights. When assigning these values, however, they should be chosen at random from either (+) or (-) values. The loss function value, which determines how well a network model is created, is a function that must approach 0 using various optimization strategies (Ivakhnenko A. G. and Lapa V. G., 1965; Fukushima K., 1980; Hinton G. E., 1986; Rumelhart D. E. et al., 1986)

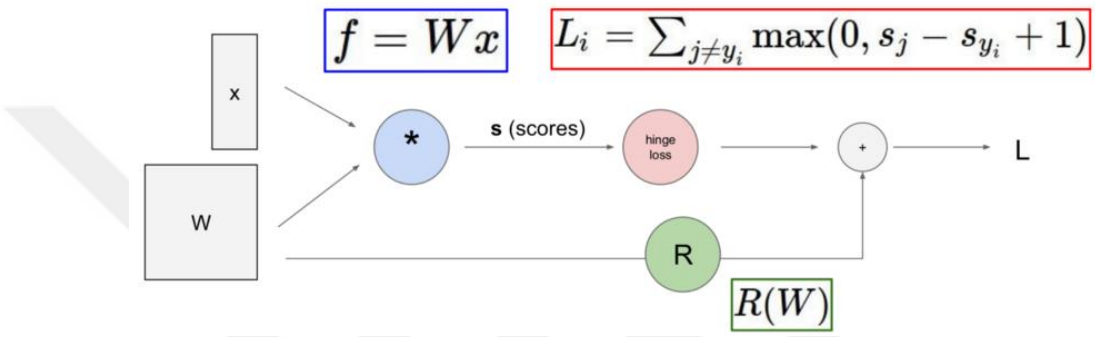

Figure 3.4: Effects of regularization on loss function in a basic artificial neural network structure (Rumelhart, D. E. et al. 1986).

The loss is never zero because of the regulation (R) value added to the loss value. The regulation value is an optional hyperparameter that has a positive influence on the positive model. The minimizing of the loss function is directly proportional to the performance. The loss function is thought to have a piecewise- linear mathematical structure (Karpathy A., 2018). As in the equations below, the loss function can be obtained. In this scenario, the method to be utilized while generating the similarity value must be decided. SVM (support vector machines) and softmax are frequently used for this (Cortes C. and Vapnik V.,1995)

$$R(W) = \sum_k \sum_l W_{k,l}^2 \tag{3.2}$$

$$L = \frac{1}{N} \sum_i L_i + \lambda R(W) \tag{3.3}$$

$$L = \frac{1}{N} \sum_i j \neq y_i [max(0, f(x_i; W)_j - f(x_i; W)_{y_i} + \Delta] + \lambda \sum_k \sum_l W_{k,l}^2 \tag{3.4}$$

3.1.2. Deep Learning

Deep learning is a sub-branch of machine learning that can provide consistently useful information when processing data in the next layer. The term "deep" refers to a set of display layers. The number of layers in the sample determines the depth of the sample. It can also be called hierarchical sign education. Deep training models may contain hundreds of successive layers. Other machine learning algorithms consist of one or two layers, commonly referred to as surface learning.

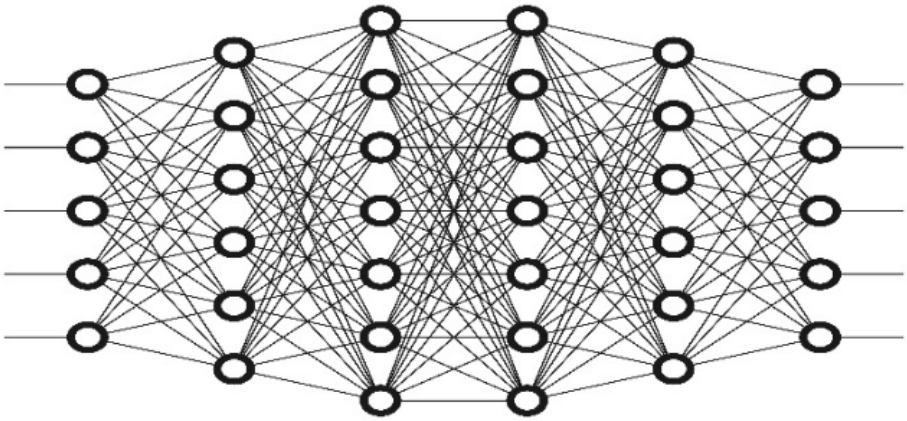

Figure 3.5: Deep Learning Network Model (Verma, A and Ranga, V., 2018)

There are many deep learning models, such as multilayer perceptrons, convolutional neural networks, recurrent neural networks, autoencoders, etc. These methods are used in many fields such as text, data, image classification, object detection, speech recognition, visual data recognition, and genomics (LeCun et al., 2015). For feature extraction and transformation, deep learning utilizes multiple layers of nonlinear processing units. The output from the previous layer is sent into the next layer (Hinton G. E., 1986). The algorithms can be supervised (like classification) or unsupervised (like pattern analysis).

In deep learning, there is a structure based on learning more than one feature level or representation of data. A hierarchical representation is created when top-level properties originate from lower-level properties. This representation learns many representation levels that correspond to various degrees of abstraction (Rumelhart, D. E. et al. 1986). A Deep Neural Network is built on the principle of learning from data. A vector of per-pixel intensity values, or characteristics such as edge clusters, and custom shapes, might be considered when a representation for an image is specified.

Some of these characteristics are more accurate representations of the facts. Deep learning approaches, rather than handcrafted features, utilize efficient algorithms for hierarchical feature extraction that best reflects the data at this level (Newborn M., 2000).

Input, hidden layers, and output are the three essential structures of multilayer sensors. The data or features extracted from the data are stored in the input layer, while the output layer indicates which class the data belongs to. The activation functions and the hidden layer enable nonlinear approaches to solve problems that cannot be addressed with linear methods. The number of hidden nodes and activation functions involved in multilayer perceptrons must be carefully considered to obtain optimum results. Below is a multilayer perceptron model.

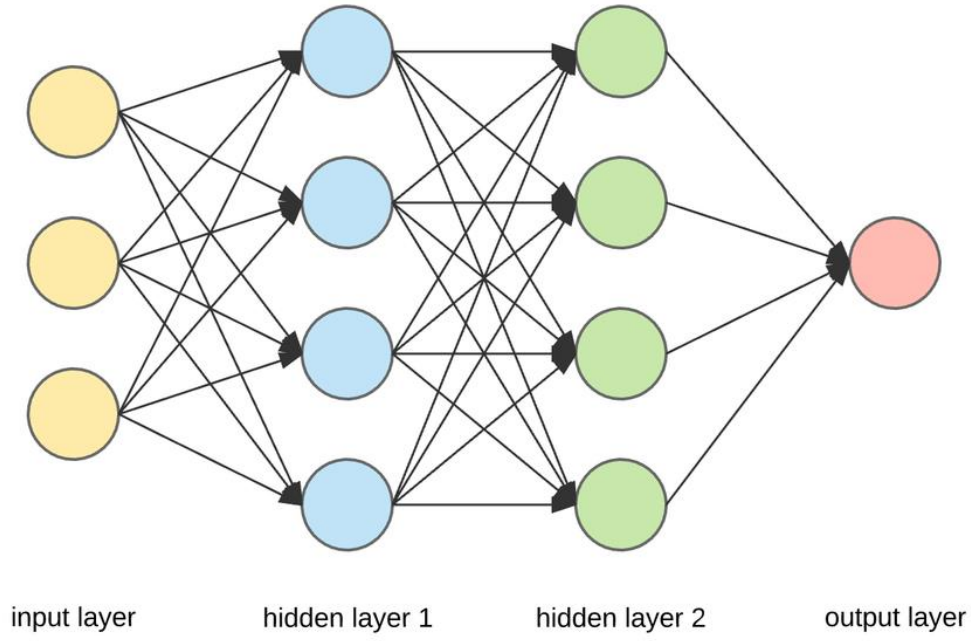

Figure 3.6: A multi-layer fully-connected neural network model

3.1.3. Convolutional Neural Networks

Convolutional Neural Networks (CNN) are multi-layer perceptron systems (MLP). Simple cells with edge-like characteristics and complex cells with larger receptors are assumed to concentrate on the entire picture, whereas cells in the visual center are split into sub-regions to cover the entire image. The visual center of animals inspired the CNN algorithm, which is a forward-looking neural network. The response of a neuron to inputs from its sensory field may be conceived of as the mathematical convolution process (Fukushima K., 1980; Hubel D. H. and Wiesel T. N., 1968; Lecun

Y. et al., 1998). A CNN consists of one or more complex layers, an inner selection layer, and one or more fully connected layers, such as a standard multilayer neural network (Lecun Y. et al., 2015).

CNN algorithms are used in a wide range of domains, including natural language processing (NLP), biomedicine, data classification, and image and sound processing. The greatest (state of the art) results were attained, particularly in the field of image processing. Cireşan et al. used CNN to lower the error rate by up to 2% in his work on the MNIST dataset (Cireşan et al. 2012). Cireşan et al. reported in another investigation on MNIST and NORB datasets that the learning process with CNN was very rapid and that it was the most successful of the approaches up to that time (Cireşan D. C. et al., 2012)

In the 2014 ImageNet Competition, all of the successful teams implemented CNN algorithms to classify and identify millions of photos and hundreds of object types (Girshick R. et al, 2014). CNN achieved effectiveness in catching faces in wide-angle ranges, including inverted faces, in 2015 research. This network was trained on 200,000 photos with faces from various angles and orientations, as well as another 20 million images without faces (Farfade S. S. et al, 2015). CNN models have been proved to be useful for a variety of tasks other than image processing. In-text categorization (Grefenstette E. et al., 2014), obtaining search queries (Shen Y. et al, 2014), sentence modeling (Kalchbrenner N. et al, 2014), classification (Kim Y., (2014), and estimation issues (R. Collobert and Weston J., 2008), excellent results were obtained. Medical research has also made use of CNN algorithms. AtomNet was the first deep neural network for medicine design, built by Atomwise in 2015(Wallach I., 2015). The technology has been used to discover novel biomolecules in illnesses like Ebola and sclerosis after being trained using 3D representations of chemical interactions (Yosinski J., 2015). CNN was also used for the Go game, and a pre-trained 12-layer CNN model defeated the traditionally developed GNU Go algorithm in 97% of the games (Maddison C. J., 2014).

Generally, CNN is constructed of layers that are categorized according to their functions; the three major types of layers are convolutional layers, pooling layers, and fully-connected layers.

Convolution Layer:

The convolution layer in a CNN is critical and fundamental building because it converts the CNN into a complicated neural network. This layer identifies essential traits by using filters or cores. The aim is to reduce the size of the dataset and only move the most significant information to the next layer. This saves a significant amount of computing labor in dense layers while maintaining good accuracy.

During forward propagation, each filter convolutions the input volume and calculates the dot product between the filter's inputs and the input at any point, using a nonlinear activation function (sigmoid, tanh, ReLU, and so on); the generated outputs are termed feature maps. The filter's responses are provided at each spatial position by the feature map (also known as the activation map). This layer's output is determined by three hyperparameters: depth, pitch, and padding.

The number of filters employed in the convolution process is represented by the depth of the output volume. On input, the stride is the number of steps the filter is moved to. Finally, padding is used to regulate the output size. **(**Bezdan, T. and Bačanin Džakula, N., 2019)

$$n_{out} = \frac{n + 2p - f}{s} + 1 \qquad (3.5)$$

Where n represents the number of filters, p represents the amount of padding, f represents the filter size, and s represents the stride.

Pooling Layer:

CNNs frequently employ the pooling layer process after convolution layers to reduce dimension, also known as subsampling or down-sampling. The pooling layer's hyperparameters indicate the filter size and strides. Pooling layers with filter size 2 and stride 2 are the most widely utilized. Pooling layers are classified into two types: max pooling and average pooling, which take the maximum and average values, respectively. Max pooling is more frequently employed than average pooling. In the pooling layer, there are no variables to train. The concept behind max pooling is that a large number indicates the possibility of detecting a feature **(**Bezdan, T. and Bačanin Džakula, N., 2019).

Fully Connected Layer:

The CNN often concludes with numerous fully connected layers after several convolutions and pooling layers. The tensor produced by these layers is turned into a vector, and then further neural network layers are added. The fully connected layers are generally the last few levels of the design; the dropout and regularization approach can be used to prevent overfitting in the fully connected layers. The architecture's last fully connected layer includes the same number of output neurons as the number of classes to be identified.

The first CNN network given in Figure 3.6 is the architecture named LeNet, which was introduced by Yann LeCun in 1988 and continued to be improved until 1998 (Srivastava, N., 2014). In the LeNet network, the lower layers consist of cascading convolution and max-pooling layers. The next upper layers correspond to the fully connected conventional MLP.

3.1.4. Recurrent Neural Network

Real-life learning systematics of people are followed in neural network architectures. There is a mental structure that consists of continually layering old information on top of new knowledge. Traditional neural networks are unable to realize this cumulative learning pattern. The most fundamental reason why standard neural networks can't solve issues like voice recognition is because of this deficiency. In speech recognition applications, for example, the former voice is significant in recognizing the following voice. The findings will be incorrect if each sound is evaluated independently. Because such a dependency cannot be described using ordinary neural networks, the Recurrent Neural Network (RNN) structure has been developed to overcome the problem (Asefisaray, B., 2018).

The figure shows the loop-permitting structure of the recurrent neural network. It takes the value "Xt" as input and outputs the value "Ht" as output. The information produced at the output of the RNN cell is used as input in the next step. Thus, it is ensured that the previous information is transferred to the next steps. The loop structure makes repetitive neural networks look complicated. However, as can be seen clearly in Figure 3.7, it is not different from the traditional neural network. It consists of a theoretically infinite number of copies of conventional neural networks. They're similar to lists and arrays. Recurrent neural networks have been used successfully in

many fields such as speech recognition, language modeling, machine translation, and image processing.

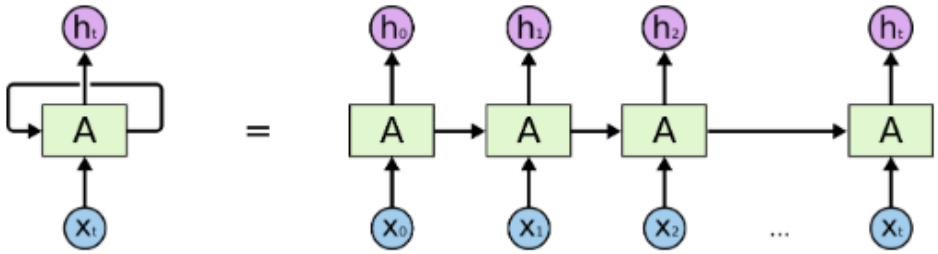

Figure 3.7: An Unrolled Recurrent Neural Network (Olah C., 2015)

Recurrent neural networks allow previous information to be used to generate new information. However, when it is important to pick just the necessary data collected before a cell, RNN structures do not allow this procedure. If improved predictions can be effectively generated with recent data, standard RNNs are adequate for this purpose. Standard RNNs, on the other hand, cannot give a solution to long-term reliance when working with issues that require much earlier knowledge due to the widening of the difference (Bengio Y. et al., 1994). To tackle these two difficulties in the RNN structure, the long short-term memory (LSTM) structure, which is a type of memory structure, has been developed. Selections are made from the prior information using this memory structure, and the required ones are passed to the next stage. It is feasible to achieve excellent results in speech recognition applications with this framework.

3.1.5. Long Short-Term Memory

LSTM is a type of recurrent neural network that can represent long-term relationships. Its purpose is to help to retain information for a long time. The neural network's job is to decide what information should be learned and to train the network. The application of this memory structure is expanding every day, and it produces good results for a variety of issues (Schmidhuber, J. et al., 2017).

Each layer in a regular RNN structure has a "tanh" function. It has a straightforward structure. The LSTM structure, on the other hand, is a chain-like and repeated set of modules. It features a unique structure with four doors instead of only one, as illustrated in Figure 3.8. These gates allow required changes in the cell state to be performed.

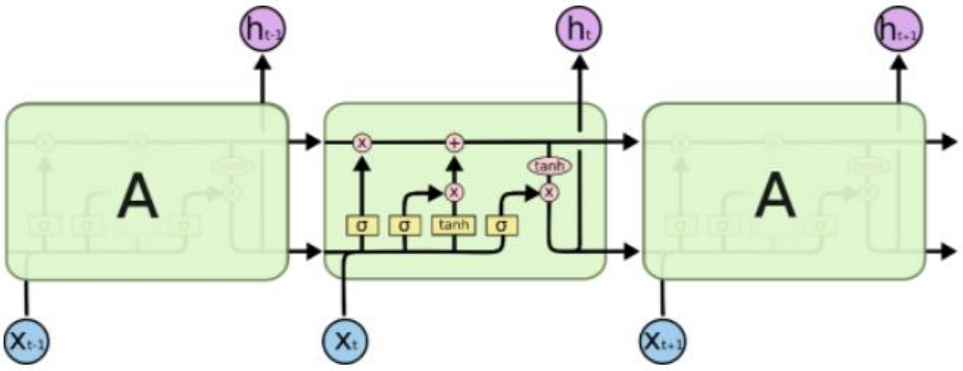

Figure 3.8: The repeating module in an LSTM contains four interacting layers. (Olah C., 2015)

Each line contains the vector that connects one node's output to the input of the next. The pink states reflect vector operations in mathematics. The learning neural network layers are indicated by yellow boxes, and forked arrows indicate that data is duplicated and delivered to multiple places. The LSTM cell functions similarly to a conveyor belt and does so throughout the cycle. The gates are used to make necessary updates, data additions, and data deletions along with the flow. The sigmoid function is used to determine which information and how much to pass. 0 implies no data can be transferred, and 1 means all data can be transmitted, according to the values produced in the range of 0-1. The LSTM structure has three of these doors.

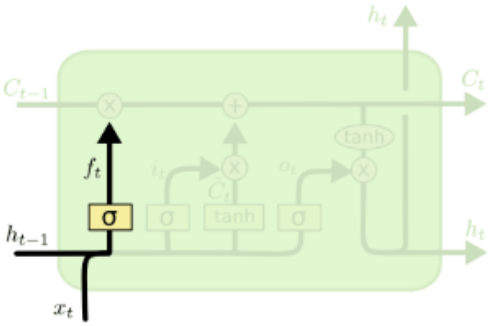

Figure 3.9: The first step in the LSTM: Forget Gate Layer (Olah C., 2015)

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{3.6}$$

The variable f_t represents the value of the gate that manages to forget the data, h_{t-1} represents the output value of the LSTM block at time $t-1$, i.e. the previous block, x_t represents the current input value, b_f represents the bias value of the forget

gate, W_f signifies the necessary weight matrix for the neurons of the f gate, and σ represents the sigmoid function. As illustrated in Figure 3.9, the initial step in the LSTM operating system is to pick which information to remove from memory. The sigmoid function is used to make the judgment that the data should be forgotten. The input values of x_t and h_{t-1} are used to generate a value between 0 and 1 for each c_{t-1} state.

In the equations, i_t is the input gate, W_i is the weight matrix for the neurons of the i gate, h_{t-1} is the output value of the previous layer, xt is the current input value, bi is the bias value of the i gate, σ is the sigmoid function, $\widetilde{C}$, It represents the candidate value prepared to update the cell state at time t.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{3.7}$$

$$\widetilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c) \tag{3.8}$$

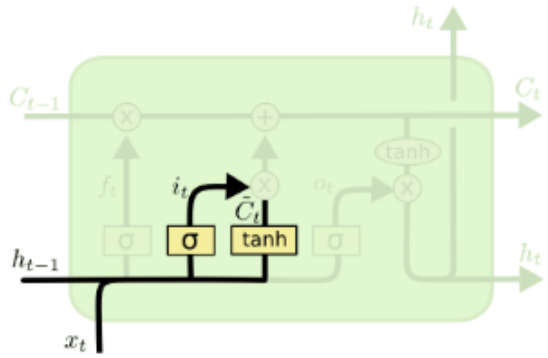

Figure 3.10: The Second step in the LSTM: Input Gate Layer (Olah C., 2015)

Figure 3.10 represents the second phase of the LSTM structure. It is decided which new information will be stored in memory during this stage. The sigmoid layer determines which values should be updated. Then, as illustrated in the above equations, a new vector of values C is formed, which may be stored using the $tanh$ layer. It's the step to decide which new value should replace which old one. Data that is no longer required should be deleted from memory after this stage. This is accomplished using the door seen in the diagram below.

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \tag{3.9}$$

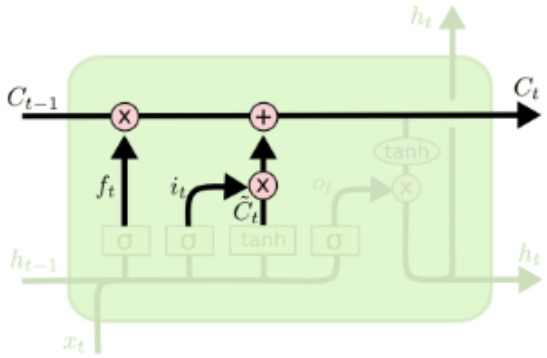

Figure 3.11: The Third step in the LSTM: Update Gate Layer (Olah C., 2015).

In the equation, C_t refers to the cell state (state of memory) at time t. As indicated in the Equation, the values to be forgotten are multiplied by the f_t value. The new C value is then added. Scaling is done on how much the past value will be modified for each new value (Olah C., 2015).

Finally, it is necessary to decide which value will be the output value, as shown in Figure 3.12. This value will of course be shaped according to the value in the cell, but the needed part will be selected. To determine which elements of the cell state to output as output, a sigmoid layer is used. A $tanh$ function is used to normalize the numbers between -1 and 1. This number is multiplied by the sigmoid function, and the result is the desired component of the state. As a result, the data that should be discarded is forgotten, but the value that should be saved remains in the memory.

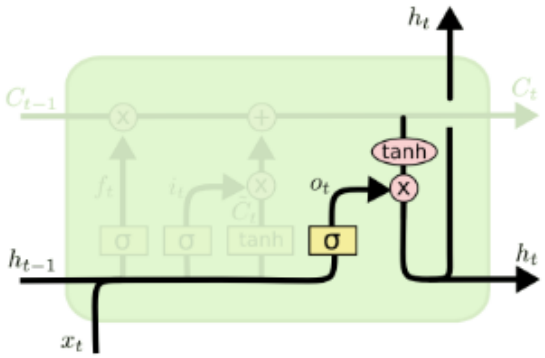

Figure 3.12: The Forth step in the LSTM: Output Layer (Olah C., 2015)

$$\sigma_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (3.10)$$

$$h_t = o_t * tanh\,(C_t) \quad (3.11)$$

σ_t is the output gate, h_{t-1} is the previous block's output, x_t is the current input value, b_o is the output gate's bias value, W_o is the weight matrix for neurons, and σ is the sigmoid function in equation 3.11.

3.2. Hyperparameters

Artificial neural network training may be done at a cheaper cost and with higher performance if hyperparameters are chosen correctly. This study's hyperparameters, which are used in machine learning techniques, are shown below.

3.2.1. Learning Rate

A ratio used in gradient minimization techniques is called the "Learning Rate." The learning coefficient ensures that slope reduction methods converge. When this coefficient is excessively big, it may not attain the target; it may wander around the global minimum point or diverge. At the same time, if the coefficient is set too low, the algorithm will take too long to reach convergence since each cycle will be made up of extremely small steps (Haykin S.S., 2009).

3.2.2. Activation Functions

The entire value was transmitted via the unit digit function in the Perceptron layers utilized in networks, and output was received. Activation functions have been designed and assembled for artificial neural networks to solve complex problems. Sigmoid and ReLU functions were applied as activation functions in this research (S. University, (2018))

Sigmoid Function: $\sigma(x) = \frac{1}{1+e^{-x}}$ (3.11)

ReLU Function: $f(x) = \max(0, x)$ (3.12)

3.2.3. Optimization Algorithms

The goal of machine learning is to produce the most accurate results with the best performance; hence it could be considered an optimization problem. Optimization

methods are frequently used in machine learning to minimize error rates and improve performance. In the solution of nonlinear problems, these strategies are employed to determine the optimum value. One of the most popular algorithms for optimizing neural networks is gradient descent (Ruder S., 2016).

3.2.3.1 Batch Gradient Descent

The Gradient descent calculates the gradient descent all over the entire data set. Memory issues come as a result of utilizing the complete dataset at once. Although it has a significant computational cost, it is more stable than other forms of gradient descent. The equation below shows how to calculate the slope of the cost function based on the parameter. (Ruder S., 2016).

$$\theta_{t+1} = \theta_t - \eta . \nabla_\theta J(\theta_t) \quad (3.13)$$

Where:

- θ is the parameter of the model
- η is the learning rate
- $\nabla_\theta J(\theta)$ is the gradient of the target function depends on the parameters.

Various optimization algorithms can be used together with these gradient descent types. The RMSProp, Adadelta, Adam, and Adamax were implemented in this research as optimization algorithms. There are variances in performance and speed between these algorithms.

3.2.3.2 Adadelta

Adadelta was designed to solve the drawback of aggressively reducing the learning coefficient in the update statement of another optimization method, Adagrad. Instead of using all of the square values of all previous slopes, Adagrad limits the quantity of value to a specific frame size w. Instead of keeping all previous slope values obtained in the w frame separately, this technique considers the squares of prior slopes as a single variable by taking their exponentially weighted average (Zeiler M., 2012).

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1-\gamma) g_t^2 \quad (3.14)$$

Where:

- $E[g^2]_t$ is the average of past slope-square values
- g_t^2 is the training square of the cost function at time t, calculated according to the parameter θ.

The learning coefficient is deducted from the equation in the Adadelta method's parameter update statement. This eliminates the need to set a default learning coefficient at the start.

3.2.3.3 RMSProp

RMSprop was developed almost concurrently with the Adadelta approach, to offer a solution to Adagrad's aggressive downsize strategy. Vertical oscillation movements are slower in the region where the cost value is decreased to acquire the minimal value rapidly; horizontal oscillation movements should be fast. The approach outlined below is implemented to achieve this speed in the RMSprop method (Tieleman T. and Hinton G., 2012)

$$E[g^2]_t = 0.9E[g^2]_{t-1} + 0.1g_t^2 \quad (3.15)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t \quad (3.16)$$

3.2.3.4 Adam (Adaptive Moment Estimation)

In addition to storing exponentially weighted averages of squares of past gradients (v_t), the adaptive moment estimation method also stores exponentially weighted averages of past gradients in momentum (m_t), as is done in RMSprop:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (3.17)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (3.18)$$

The Adam algorithm's developers observed that the initial values of m_t and v_t, which were specified as the zero vector, lean-to zero, especially in the early iterations and when adjacent to β_1 and β_2 (Kingma, D. and Ba, J., 2014).

They performed polarization-corrected first and second-moment calculations to overcome this issue:

$$m_t' = \frac{m_t}{1 - \beta_2^t} \quad (3.19)$$

$$v_t' = \frac{v_t}{1-\beta_2^t} \tag{3.20}$$

The t value in the β_i^t $(i = 1, i = 2)$ expression, which is also included in the equations, is considered a power. The parameters are then updated using these two expressions, just as they were in Adadelta and RMSprop.

As a result, the Adam update equation is as follows.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{v_t+\epsilon}} m_t' \tag{3.21}$$

The default values for this rule are 0.9 for β_1; It is specified as 0.999 for β_2 and 10^{-8} for ϵ.

3.2.3.5 Adamax

In this method, Adam's update method rule's factor v_t adjusts the gradient inversely to the previous gradients' l_2 norm (through the term v_{t-1}) and the current gradient (g_t^2) (Kingma, D. and Ba, J., 2014).

The optimum values for the algorithm are; the learning rate was determined by the developers of the algorithm as 0.002 for η, 0.9 for β_1 and 0.999 for β_2.

3.2.4. Epoch, Iteration, Batch Sizes

Epoch means that the entire data set travels back and forth across the entire network. Iteration is the back-and-forth transfer of data of a given batch size over the network. The batch size is the amount of data received from the forward and reverse spread datasets.

The larger the batch size, the more memory space is required. If you choose a large batch, the training sample for each iteration will also be large. If a dataset of 1000 data is specified with a batch size of 500, two iterations are required to complete the epoch.

A higher epoch number does not mean that the mesh will provide more accurate results. Typically, these two parameters are used to measure the change in value at each step (for example, top 1 accuracy, the margin of error, and weight change). You et al. You Y. et al. created Figure 3.13 showing the correlation between epoch and accuracy of the top 1 test. In this way, regardless of the number of epochs and other hyperparameter values, one can observe that after a point, the accuracy gain is slower

and the accuracy stops increasing. The same situation was observed in the studies of Hoffer (Hoffer E. et al., 2017). The decrease in validation error started after the 5000th iteration and slowed down after the 20000th iteration.

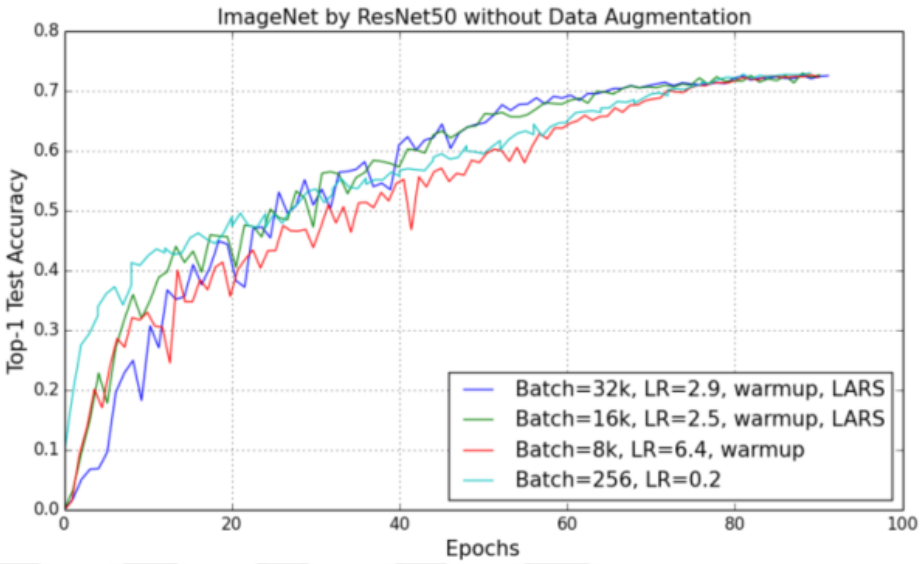

Figure 3.13: Test Accuracy change by Epoch (You, Y., 2017)

The batch size can also affect network accuracy. A study by Goyal et al. (2017) shows that 8192 or 256 batch measurements gave very accurate values. Excessive package size increased Top-1 validation errors too fast. Similar to this result, You et al. (2017) ResNet50 concluded that accuracy increases when the package size exceeds 10,000 (You Y. et al, 2017).

3.3. Evaluation Methods

This section describes performance standards for evaluating traditional approaches.

3.3.1. Mean Absolute Error (MAE)

The mean absolute error measures the distinction between two continuous variables. MAE stands for the average vertical distance between both the true value and the best-fitting vector. The MAE value is commonly utilized in regression and time series problems because it is simple to interpret. The MAE is a linear score that examines the mean size of failures in a group of predictions regardless of evaluating their direction and scores all individual errors equally on the mean. The MAE value

might be anywhere between 0 and ∞. Scores that are negatively oriented, or have lower values, perform better.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |x_i - x| \quad (3.22)$$

x_i is the measurement, x is the true value, n is the number of errors in this formula (Wang W. and Lu Y., 2018).

3.3.2. Mean Squared Error (MSE)

The MSE is a measure of how near the regression line is to a series of parameters. This is accomplished by squaring the distances between the points and the regression line (the distances are the "errors"). Squaring is necessary to remove any negative signals. Likewise, it proposes extra weight to huge differences. It's known as the mean squared error as discovering the average of a set of errors. The mean squared error is defined as finding the average of a set of errors. The stronger the forecast, the lower the MSE (Fürnkranz J. et al., 2010)

The steps of the calculation mse are as follows;

Firstly, finding the regression line is needed. Then, X values must be inserted into the formula for linear regression to see the new Y values (Y'). After that, subtracting the new Y value from the original to get the error must be done. Finally, the errors must be squared and added up (the Σ in the formula is summation notation). The last calculation finds the mean.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2 \quad (3.23)$$

n is the number of items, y_i is the actual value and $\tilde{y}_i$ is the forecast value in this
function.

3.3.3. Mean Absolute Percentage Error (MAPE)

The mean absolute percentage error is one of the accuracies measuring methods used with forecast systems (MAPE). It is determined as the average absolute percentage error throughout each time period minus actual values divided by real values and expressed as a percentage.

The mean absolute percent error is widely used to quantify the accuracy of predictions in regression and time series models. MAPE cannot be calculated if the

real values contain zeros since division by zero will occur. For very low forecast values, the percentage error cannot exceed 100 percent, while there is no maximum limit for very high projected values. When MAPE is used to compare the accuracy of estimators, it is skewed since it chooses a technique with very low estimates regularly. An accuracy metric that finds the ratio of projected values to actual values can solve this modest but important problem. This method yields estimates that can be explained using the geometric mean. (Kim S. and Kim H., 2016)

$$MAPE = \frac{1}{n}\sum_{l=1}^{n} \left| \frac{y_i - \tilde{y}_i}{y_i} \right| \quad (3.24)$$

n is the number of fitted points, y_i is the actual value and $\tilde{y}_i$ is the forecast value in
this function.

3.3.4. Confusion Matrix

The confusion matrix is a common measurement for solving classification problems. It can be used for both binary classification problems and multiclass classification problems.

Table 3.1: Confusion Matrix

	Negative (Predicted)	**Positive (Predicted)**
Negative (Actual)	TN	FP
Positive (Actual)	FN	TP

The counts of predicted and actual values are displayed using confusion matrices. True Negative TN is the output, and it shows the number of correctly detected negative situations. Similarly, TP stands for True Positive, which indicates the number of positive cases accurately detected. False Positive and False Negative are abbreviated as FP and FN respectively. FP denotes the number of real negatives instances classified as positive, whereas FN denotes the number of real positive cases classified as negative.

3.3.4.1. Accuracy

When it comes to categorization, accuracy is one of the most important factors to consider. The accuracy of a model is calculated using the formula below (through a confusion matrix).

The calculation of accuracy is represented as;

$$Accuracy = \frac{TP+TN}{TN+FP+TP+FN} \quad (3.25)$$

3.3.4.2. Precision

Precision is defined as the ratio of true positives to the sum of true positives and false positives. Precision focuses on how frequently false positives were included in the sample. The model is completely accurate if there are no false positives (FPs). The more FPs in the composition, the lesser the accuracy. The confusion matrix's positive and negative values are used to calculate a model's accuracy. The calculation of precision is represented as;

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \quad (3.26)$$

3.3.4.3. Recall

Instead of focusing on the number of false positives predicted by the model, recall considers the number of false negatives in the mix. The recall rate suffers when a false negative is expected. The equations are also completely different since the penalties for accuracy and memory are opposite. The calculation of recall is represented as;

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \quad (3.27)$$

3.3.4.4. F-Measure

Precision and recall are harmonically equal. It takes both false positives and false negatives into account. As a consequence, it handles an imbalanced dataset effectively. Recall and precision are equally represented in the F Measure. The calculation of F Measure is represented as;

$$F - Measure = \frac{2*Recall*Precision}{Recall+Precision} \quad (3.28)$$

3.4. Materials

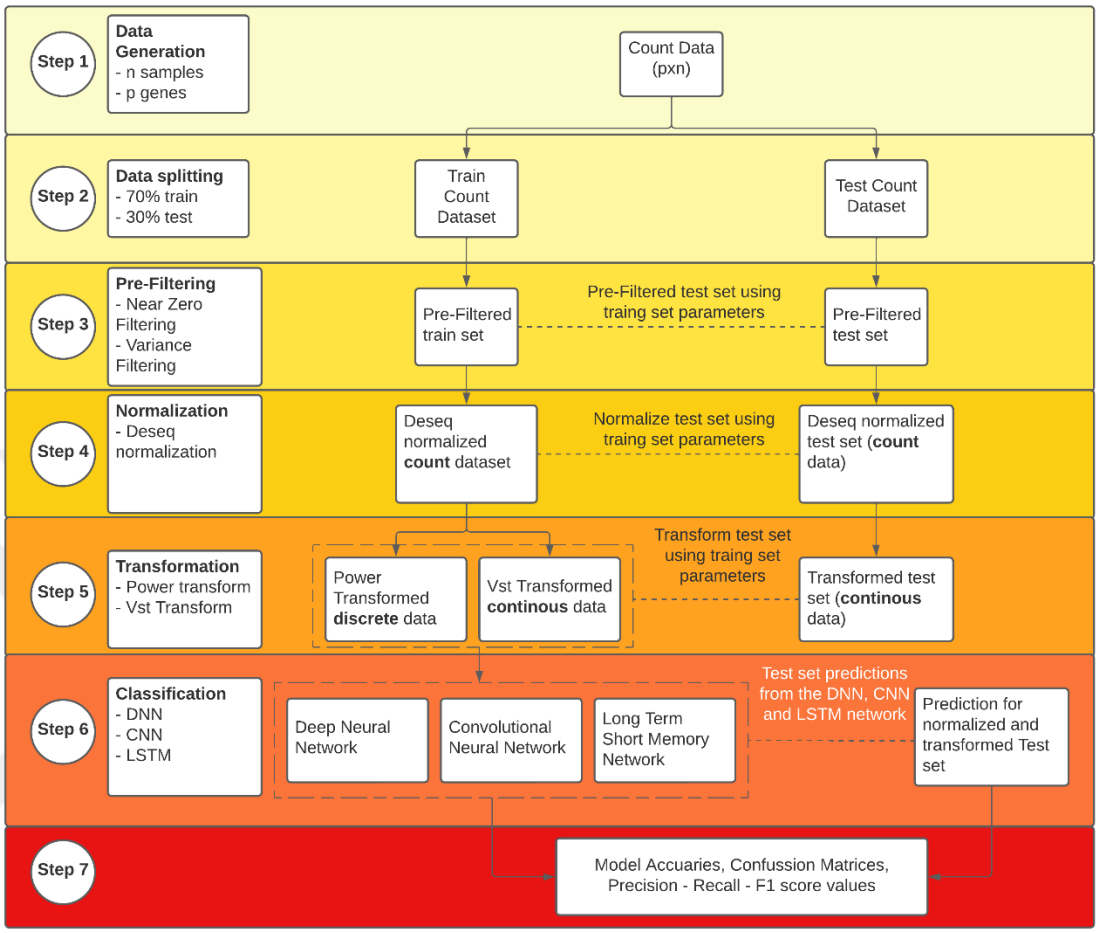

Figure 3.14: Simulation Step of the System

Pre-processing the data to be used in data classification studies is important to increase the performance of the study (Kotsiantis S. et al., 2006). In this study, classification is performed on two real RNA-SEQ datasets. Cervical cancer data which is collected by Witten is the first data set. Each cervical data set consists of a human sample of 58 (29 non-tumor (N) and 29 tumors (T)) using Solexa/Illumina platform (Witten, 2011). The goal of this work was to find genes that were differently expressed in healthy and tumor samples, as well as new miRNAs linked to cervical cancer. The analysis is based on a count matrix with 58 samples and 714 miRNAs. The secondary set of real data, on the other hand, was made by integrating Alzheimer's data gathered by Leidinger et al (2013). This dataset contains 2,801 miRNA sequencing reads extracted from blood samples of 48 Alzheimer's patients and 22 age-matched controls.

The patients were chosen from a group of people who had been diagnosed using the Alzheimer Disease Assessment Scale cognitive subscale (ADAS-Cog), the Wechsler Memory Scale (WMS), the Mini-Mental State Exam (MMSE), and the Clinical Dementia Rating Scale (CDRS) (CDR). The researchers wanted to see whether any miRNAs were linked to Alzheimer's disease and other neurological illnesses. Alzheimer's sets are samples of 49 persons (15 control (C) and 34 Alzheimer's disease (AD)).

Table 3.2: Description of Datasets

Dataset	*Number of Samples*	*Number of Genes*	*Provided Platform*
Cervical Cancer	50 different set x 58 sample = 2900	714 miRNAs	Solexa/Illumina platform
Alzheimer	50 different set x 49 sample = 2450	2801 miRNAs	Selected subjects were diagnosed through several tests: ADAS-Cog, WMS, MMSE, CDR

Also, the sets are separated as 70% training data and 30% test data. The number of genes for the datasets differs according to the filtering processes applied during the analysis. For this reason, when examining the effect of the number of genes selected during the training stage, 5%, 10%, and 30% of the genes in the related data set will be done. The decimal number will be rounded down when determining the number of genes (24.7 genes will be selected as 24 genes).

As represented in Step 3 Near-Zero Filtering and variance filtering methods are utilized for the Pre-Filtering process. These two methods are applied to train and test datasets. After the method is applied, pre-filtered test sets are using training set parameters. For the normalization step which is step 4, the Deseq normalization method is applied. The prefiltered sets are normalized and ready to be transformed as the next step. Same as the pre-filtering step, normalize test sets using training set

parameters. Next, transformation is applied to make the data set in the application form classified. Transformation methods are Power and VST transformation, power transformation generates discrete data but VST transformation generates continuous data. The transformed test set uses training set parameters as this method applied before. After the data set are ready, three classification methods are utilized to get necessary results which are accuracy, confusion matrix result, precision, recall, F1 score values, and standard deviations.

CHAPTER 4

RESULTS AND DISCUSSIONS

In this section, the experimental results related to the gene selection method and classification of the cervical cancer dataset and the Alzheimer's dataset are completed. As mentioned in the previous sections, the path determined in the selection of the data set is as follows: 5%, 10%, and 30% of the genes in the relevant datasets were selected and three different classification algorithms were used to classify these selected gene subsets. The genes selected here were associated with the disease, and the most relevant part was selected while making the percentage selection.

In this study, the training and testing phase were completed with the code written in python 3.9 with Deep Neural Network, CNN, and LSTM models as in most of the studies in the literature (Priyadarshini, I. and Cotton, C., 202; Ganda R. and Mahmood A., 2018). Keras (Chollet F. et al., 2015) which is a deep learning library for Python that provides a convenient way to define and train almost any type of deep learning model was used to create learning models, adding layers with activation functions and padding, adding hyperparameters, evaluating the parameters, predict tests' values output, and finally receive the result. Keras is a high-level neural networks API written in Python that can run on TensorFlow, Theano, and CNTK. Keras is utilized with TensorFlow GPU in this research. Thanks to the many functional functions it contains, Keras allows us to easily create and train a Deep Neural Network model. For this reason, Keras is one of the libraries recommended for those who are new to deep learning.

The technical specifications of the machine which is used for this study are as the processor is Intel(R) Core (TM) i7-8750H CPU 2.21 GHz, 16 GB RAM The Graphic card is GTX 1660 TI, Operating System is Windows 11 Pro version 21H2. Using IDE is PyCharm 2021.3.3 Professional Edition for Education Only.

As mentioned, Python was used in this study. The first part of the code, reading values from the files and modifying them to make the same format to prevent the models from throwing errors. There are 50 training and test data sets in excel files available for Cervical separately. In the same way, there are 50 training and testing data sets in excel files available for Alzheimer's. While the values were obtained from these excel files, the genes to be used for training were given to the model separately

in such a way that they are 5%, 10%, and 30% of the number of genes. Then, some hyperparameters are specified: Adam, Adamax, Adadelta, and RMSProp implemented as optimizers. The learning Rate of these optimizers was 0.001, 0.005, and 0.01. The number of Epochs was 50,75, and 100 with three batch sizes which are 32,64, and 128. Keras DNN, CNN, and LSTM models are determined and their architectures of them are visualized in the 4.1 Network architectures sections. The training and evaluation for tests were done with Deep Neural Network, CNN, and LSTM model for each excel file of the Cervical Cancer dataset and Alzheimer dataset with 4 optimizers, 3 learning rates, 3 different number of epochs, and 3 different batch sizes for 5%, 10% and 30% of the number of genes. Almost 32.400 training outputs were generated and these processes took about 245 hours. After the results were ready, the average values of a metric which are loss, mae, mape, accuracy, precision, recall, and f1 score were calculated to show the change in their performance instead of a single value.

4.1 Network Architectures

A detailed representation of the architectures of the sorting algorithms mentioned in the previous section that was created using the Keras model is presented in this section. The presentations show the added layers to Keras models of the classification algorithm.

The first graph represents the deep neural network architecture. There were 3 Dense layers, two of them have ReLU as activation function, and the last dense layer has sigmoid as activation function.

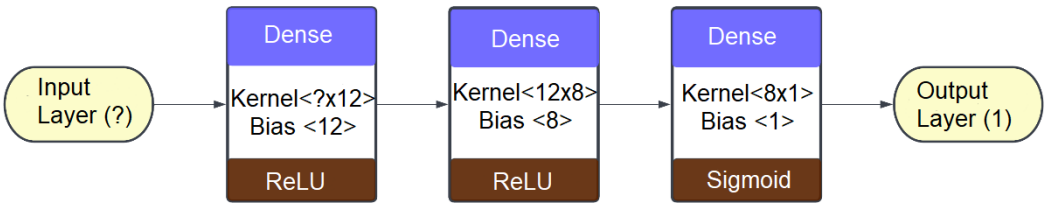

Figure 4.1: Architecture of Deep Neural Network Model

The second graph represents the Convolutional Neural Network architecture. The first layer after the Input layer is Spatial Dropout 1D before Convolutional Layer

with ReLU as activation function. Then Max pooling (3) layer is coming. Same as before, Convolutional Layer with ReLU as activation function and max-pooling (3) layers are embedded. After that, a flattened layer is added, then two Dense layers are placed with ReLU and Sigmoid activation functions before the Output Layer.

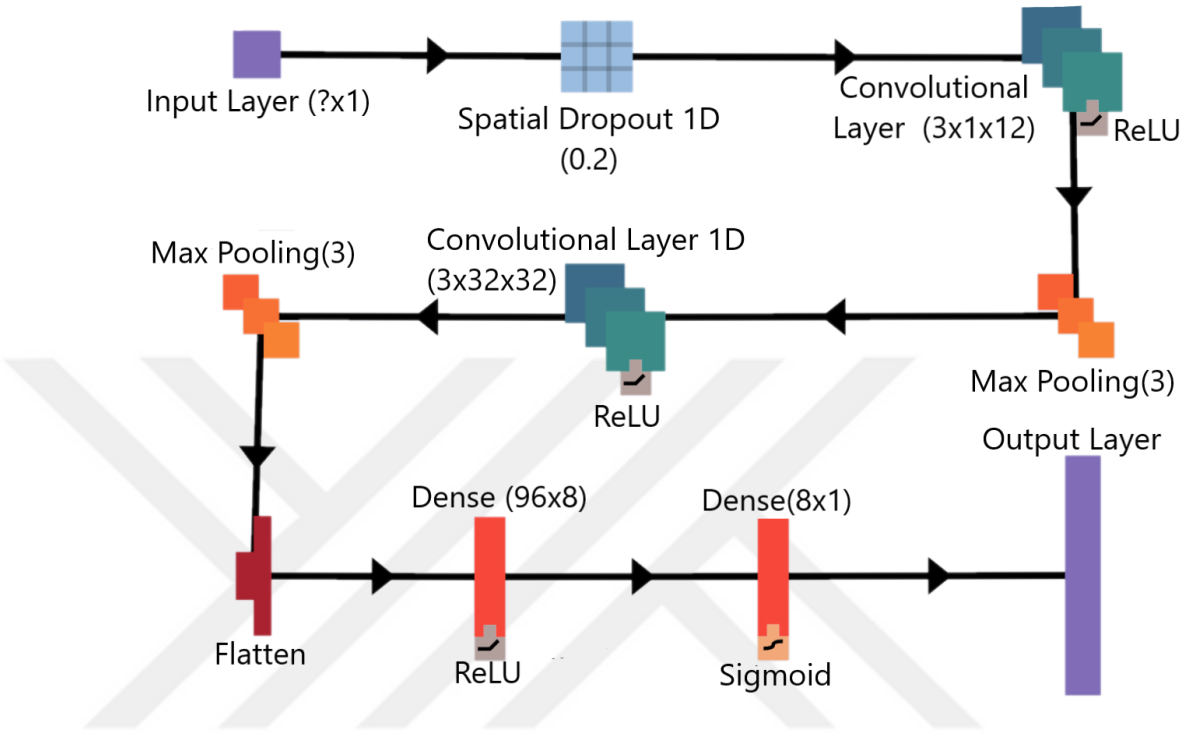

Figure 4.2: Architecture of Convolutional Neural Network Model

The last graph represents the Long Short-Term architecture. The first layer is the Convolutional Layer with ReLU as an activation function. Then Max pooling layer is added. After that, two LSTM layers are embedded. Then two Dense layers are placed with ReLU and Sigmoid activation functions before the Output Layer as you can see in Figure 4.3.

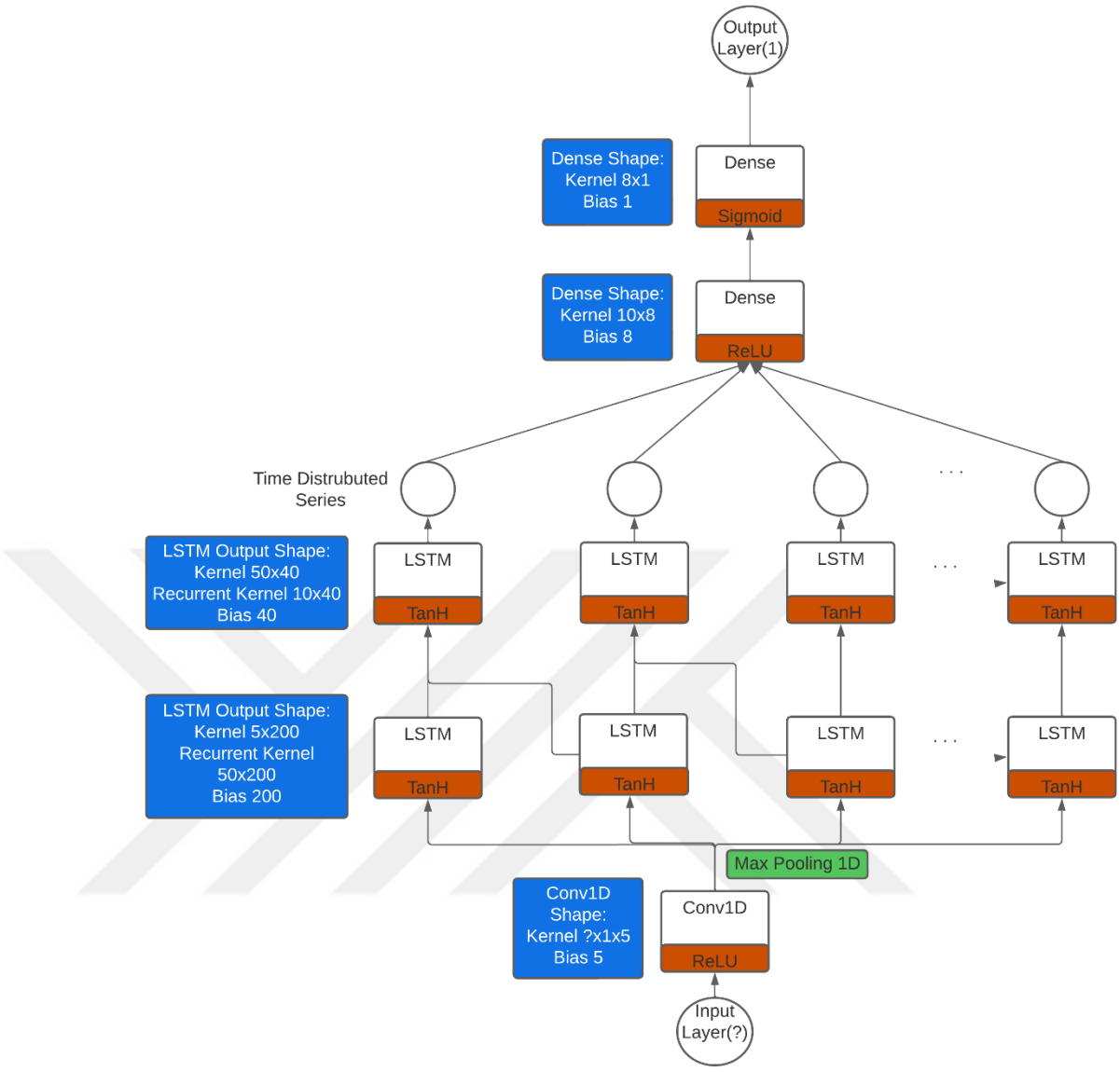

Figure 4.3: Architecture of Long Short-Term Memory Model

4.2. Experimental Results

In this section, the tests performed in this thesis and their results will be discussed. As stated in the previous sections, 3 different classification algorithms and 4 different optimization algorithms with 3 different learning rates are trained differently with 3 different batch sizes and 3 different epoch numbers. The three classification algorithms mentioned were trained in 108 different ways and these 50 Cervical Cancer were applied to the Data Set and Alzheimer's Data Set separately. These training has been repeated separately in such a way that 5 percent, 10 percent,

and 30 percent of the genes are selected. In summary, in this study, the system was trained 972 times for each data set. The total number of training results produced is 97,200 units. In a model trained here, test data were also tested and related results were obtained. The different methods are represented in Table 4.1.

Table 4.1: Training variables

Selected Percentage of the Genes	5%	10%	30%	
Classification Algorithms	DNN	CNN	LSTM	
Optimization Algorithms	Adam	Adamax	Adadelta	RMSProp
Learning Rates	0.01	0.005	0.001	
Number of Epochs	50	75	100	
Batch sizes	32	64	128	

4.2.1 Case 1: 5% Gene Selection

The first training method is done by the selection of 5% of related genes. The number of selected genes is 36 for Cervical Cancer Dataset. This number is 141 for Alzheimer's data set. The selected genes are different for each data set, as mentioned there are 50 datasets for Cervical cancer, and 50 datasets for Alzheimer's disease. For example, while the most related 5 genes from the first cervical dataset are respectively miR-143, miR-10b*, miR-944, miR-1, and miR-140-5p, for the second dataset, they are miR-10b*, miR-205, miR-147b, miR-542-5p, miR-944 respectively. After the selection is done for both Cervical and Alzheimer's datasets, training was processed. There were 108 results for each classification method and each disease. After all the training was completed and results are gathered, the average of loss, mae, mape, accuracy, precision, recall, and f1 score values are calculated for each classification method and each disease.

The average cervical cancer datasets results are represented in Table 4.2.

Table 4.2: Cervical Cancer dataset 5% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	**0.048**	**0.057**	**9.089**	**94.94%**	**0.990**	**0.916**	**0.948**
	0.044	**0.060**	**12.185**	**94.94%**	**0.986**	**0.920**	**0.949**
CNN	0.073	0.152	54.262	92.12%	0.971	0.888	0.921
LSTM	0.142	0.181	43.086	83.41%	0.939	0.741	0.794

Table 4.2 represents the most accurate (highest accuracy) results for each classifier. Also, the loss, mae, mape, accuracy, precision, recall, f1 score, and standard deviation results are included in Table 4.2. The reason the Deep Neural Network classifier has two results is their optimizers and batch size are different. The first column's hyperparameters are "Adam" optimizer, "0.005" learning rate, epoch number is "75", and the batch size is "128". The hyperparameters of Deep Neural Network's second results from the table are "RMSProp" optimizer, "0.005" learning rate, epoch number is "75", and the batch size is "32". As is seen in Table 4.2 the most accurate results belong to Deep Neural Network classifiers. CNN classifier has the closest accuracy result. The hyperparameters of this classifier are "Adam" optimizer, "0.001" learning rate, epoch number is "50", and the batch size is "32". LSTM classifier has the lowest accuracy with the hyperparameters which are "RMSProp" optimizer, "0.01" learning rate, epoch number is "100", and the batch size is "128".

The Adam and RMSProp optimizers, according to the data, contributed to the best outcomes at 5% gene selection. Obtaining the greatest results with these optimizers required 128 and 32 batch sizes, as well as a 0.005 learning rate for 5% gene selection. As shown in the table, when the accuracy value is greater, the parameters associated with error calculation, such as Loss, Mae, and MAPE, are lower, however, precision, recall, and F1 score values, reflect the validity of the result, and are high in exact correlation to accuracy. To conclude, the numbers in the table indicating the highest accuracy value's error values are lower, while the values displaying the system's performance are higher.

To obtain the confusion matrices, accuracy, and loss graphs of these results, training was performed one more time with the hyperparameters where the classifier gave the highest performance on a randomly selected data set.

Table 4.3: 5% Gene-selected Cervical Cancer Deep Neural Network Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	0
	Non-Tumor	0	9

The first retrain was performed for Deep Neural Network with the higher result's hyperparameters which are "RMSProp" optimizer, "0.005" learning rate, epoch number is "75", the batch size is "32". The confusion matrix result was promising for this training. The accuracy and loss graphs are represented in Figures 4.5 and Figure 4.6. In this tutorial, the accuracy was 100%, and the system correctly estimated the output of the test values without making any mistakes. Likewise, Recall, Precision, and F1 Score values were 1.

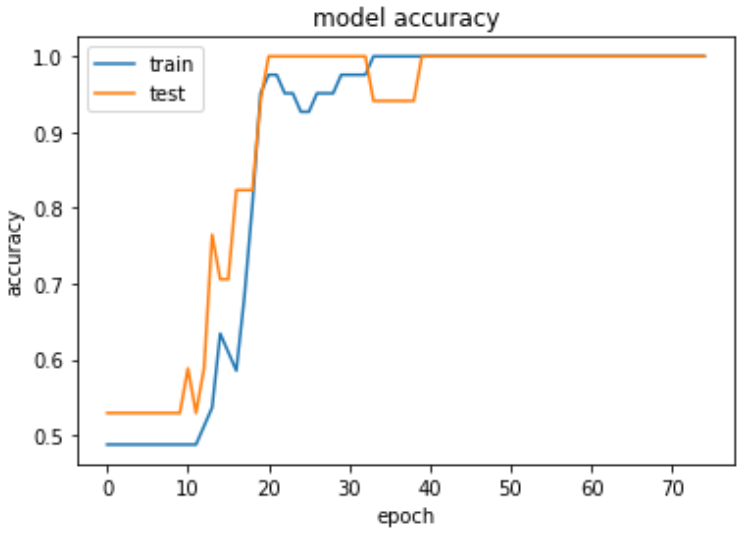

Figure 4.4: Model accuracy for Deep Neural Network 5% Gene-selected Cervical Cancer.

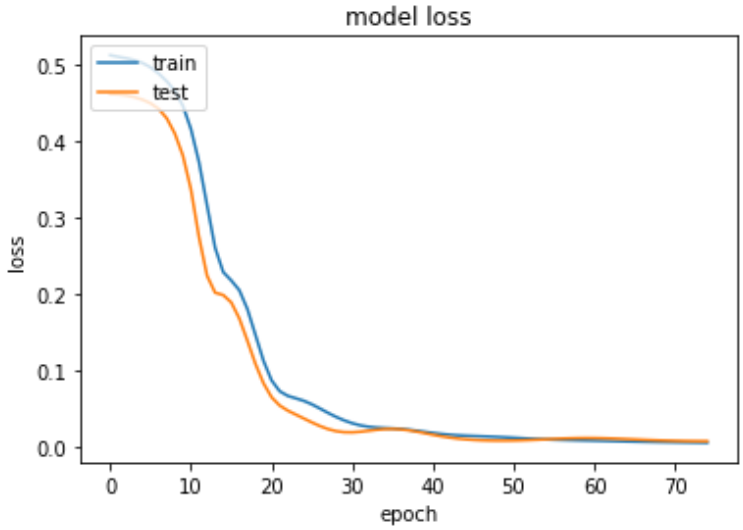

Figure 4.5: Model loss for Deep Neural Network 5% Gene-selected Cervical Cancer.

After retrain was performed for CNN with the hyperparameters of this classifier are “Adam” optimizer, “0.001” learning rate, epoch number is “50”, the batch size is “32” for another randomly selected dataset. The same as the Deep Neural Network outcomes for this training, the confusion matrix result was encouraging as well.

Table 4.4: 5% Gene-selected Cervical Cancer CNN Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	0
	Non-Tumor	1	8

The accuracy and loss graphs are represented in Figures 4.7 and Figure 4.8. In this tutorial, the accuracy was 94.11%, and the system estimated the output of the test values by making only 1 mistake. Likewise, the Recall value was 1, and Precision and F1 Score values were close to 1.

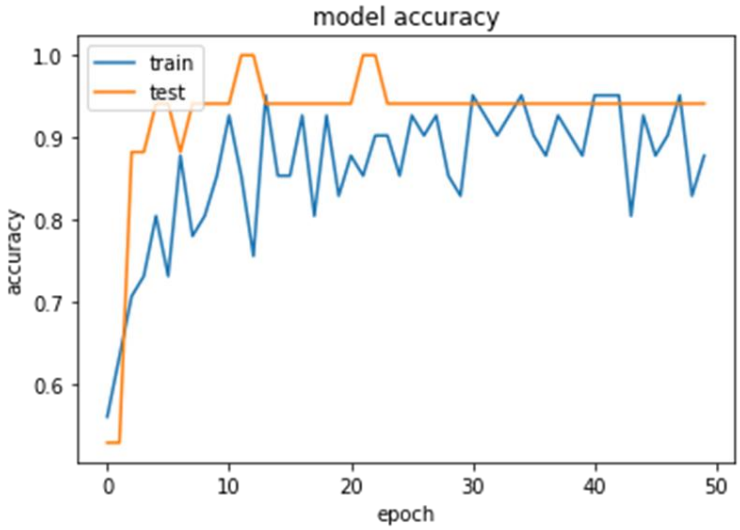

Figure 4.6: Model accuracy for CNN 5% Gene-selected Cervical Cancer.

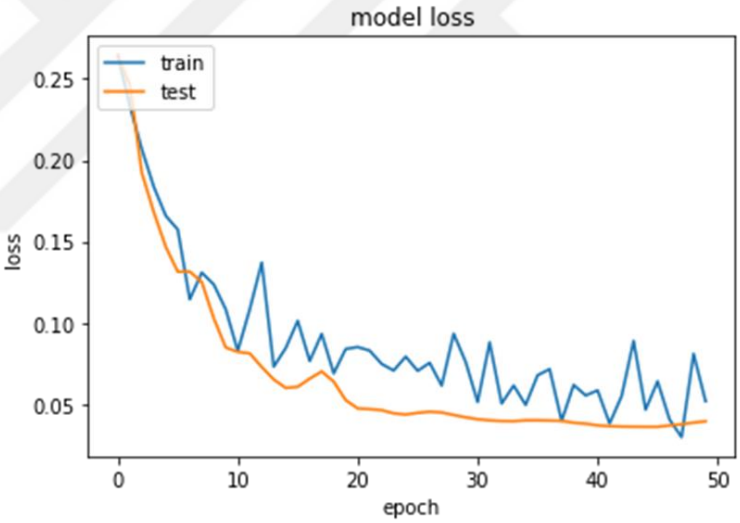

Figure 4.7: Model loss for CNN 5% Gene-selected Cervical Cancer.

The last retrain was processed with LSTM for 5% Gene-selected Cervical Cancer, again with a randomly selected dataset. The results of LSTM's confusion matrix, which is lower than the other two systems, also appear acceptable. The confusion matrix is displayed in Table 4.5.

Table 4.5: 5% Gene-selected Cervical Cancer LSTM Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	6	2
	Non-Tumor	0	9

Figures 4.9 and 4.10 show the accuracy and loss graphs, respectively. The accuracy in this training was 88.23%, and the system predicted the output of the test values with two errors. The Precision value was 1, while the Recall and F1 scores were lower.

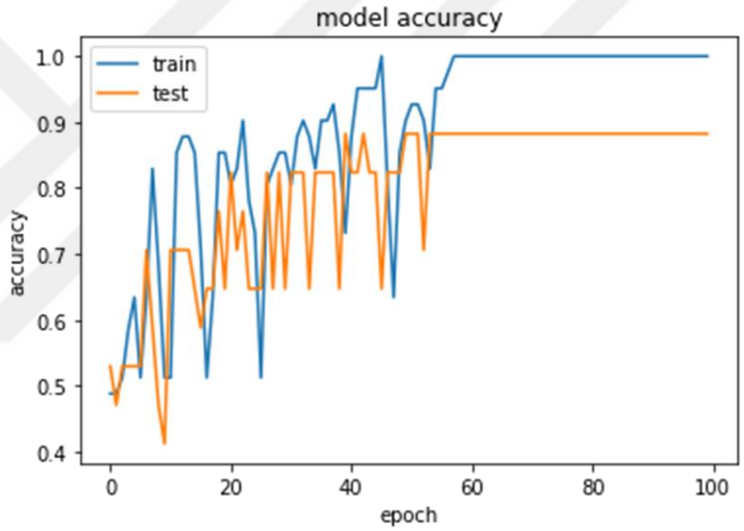

Figure 4.8: Model accuracy for LSTM 5% Gene-selected Cervical Cancer.

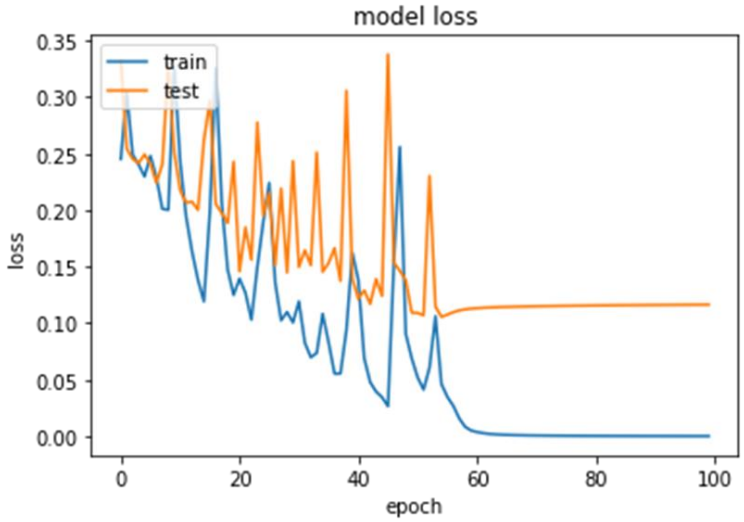

Figure 4.9: Model loss for LSTM 5% Gene-selected Cervical Cancer.

After the Cervical Cancer training was completed and the outcomes of the retraining were obtained, Alzheimer's training was successfully performed. The average of the Alzheimer's dataset's results is shown in Table 4.6.

Table 4.6: Alzheimer's disease dataset 5% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	0.126	0.255	134.423	83.89%	0.860	0.946	0.893
CNN	**0.110**	**0.238**	**125.322**	**84.74%**	**0.875**	**0.928**	**0.895**
LSTM	0.147	0.247	131.455	80.05%	0.858	0.898	0.860

Table 4.6, like Table 4.2, shows the most accurate (highest accuracy) results for each classifier, as well as the loss, mae, mape, precision, recall, f1 score, and standard deviation values.

The alternative hyperparameters were more successful for Alzheimer's. CNN was the most accurate classifier, with the following hyperparameters for the results from the table: "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "128" batch size. The Deep Neural Network classifier has the closest accuracy result, as shown in table 4.6. This classifier's hyperparameters were " Adamax " optimizer, "0.01" learning rate, "100" epoch number, and "128" batch size. With the hyperparameters "Adam" optimizer, "0.005" learning rate, "100" epoch number, and

“128" batch size, the LSTM classifier has the lowest accuracy as it was in the cervical dataset.

According to the statistics, the Adamax optimizer led to the best results for the Alzheimer data set at 5% gene selection. For this illness, 128 batch sizes and a 0.01 learning rate were necessary to get the best outcomes with these optimizers for this dataset. With the highest epoch number, the system also performs better. As previously stated in the cervical section, as the accuracy value increases, the parameters associated with error calculation, such as Loss, Mae, and MAPE, decrease; however, precision, recall, and F1 score values, which reflect the validity of the result, increase in direct proportion to accuracy. The error values in the table representing the greatest accuracy value are lower, but the values reflecting the system's performance are greater, as previously.

The training was repeated using the hyperparameters where the classifier showed the best performance on a randomly selected data set to acquire the confusion matrices, accuracy, and loss graphs of these findings like it was processed for the cervical cancer dataset.

For 5% Gene-selected Alzheimer's disease, the initial retrain was evaluated using a Deep Neural Network. The confusion matrix of the Deep Neural Network model has satisfactory accuracy. The confusion matrix may be seen in Table 4.7.

Table 4.7: 5% Gene-selected Alzheimer Deep Neural Network Confusion Matrix

		Predicted Values	
		Alzheimer’s Diseased	Control
Actual Values	Alzheimer’s Diseased	9	1
	Control	0	2

The accuracy and loss graphs are shown in Figures 4.11 and 4.12, respectively. This training was 91.67 percent accurate, and the system correctly predicted the test values with only one failure. Recall and F1 scores were 0.9 and 0.95, respectively, while Precision was 1.

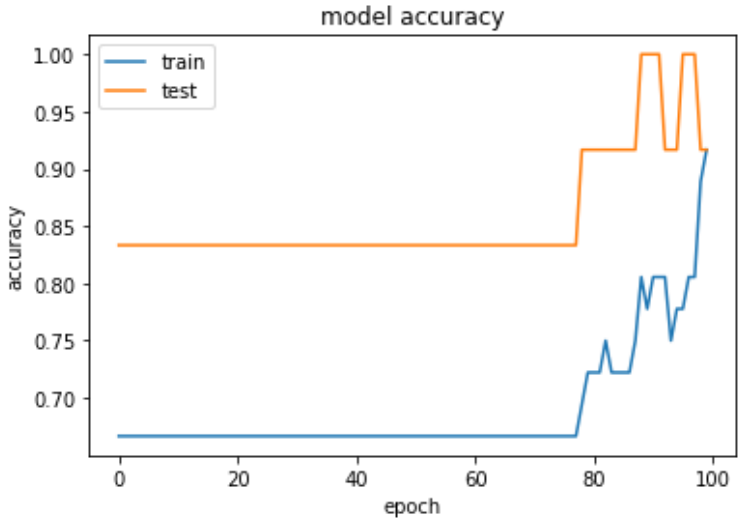

Figure 4.10: Model accuracy for Deep Neural Network 5% Gene-selected Alzheimer's.

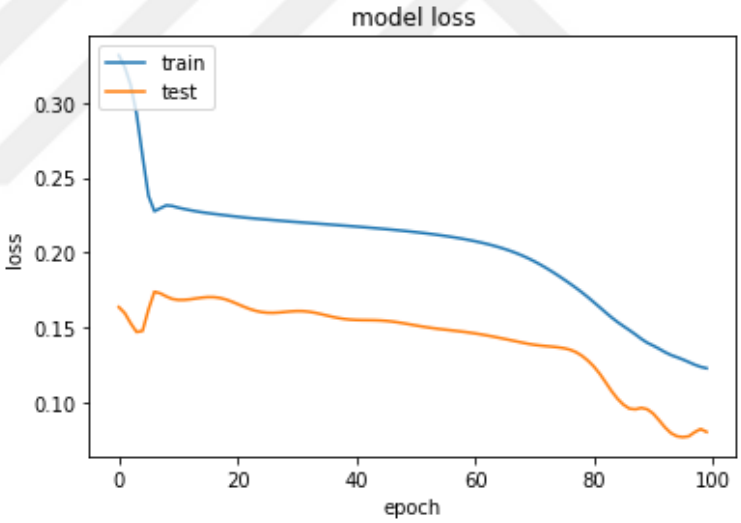

Figure 4.11: Model loss for Deep Neural Network 5% Gene-selected Alzheimer's.

The second retrain was examined using CNN, which provides the highest accurate outcome, for the percent 5 Gene-selected Alzheimer's disease. The CNN model's confusion matrix has the same results as the Deep Neural Network retrain confusion matrix outcome. Table 4.8 displays the confusion matrix.

Table 4.8: 5% Gene-selected Alzheimer CNN Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	9	1
	Control	0	2

This retraining was 91.67 percent accurate, and the model accurately predicted the test values with only one error, similar to Deep Neural Network outcomes. Precision was 1, while recall and F1 values were 0.9 and 0.95, respectively. Figures 4.13 and 4.14 indicate the accuracy and loss graphs, respectively.

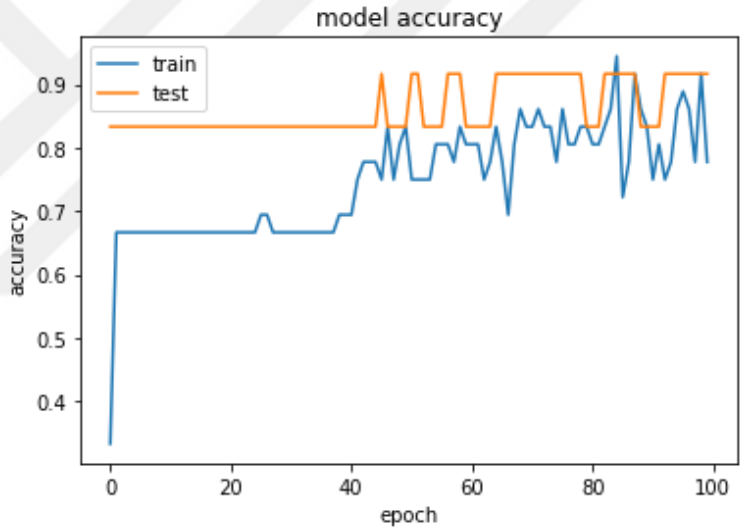

Figure 4.12: Model accuracy for CNN 5% Gene-selected Alzheimer's.

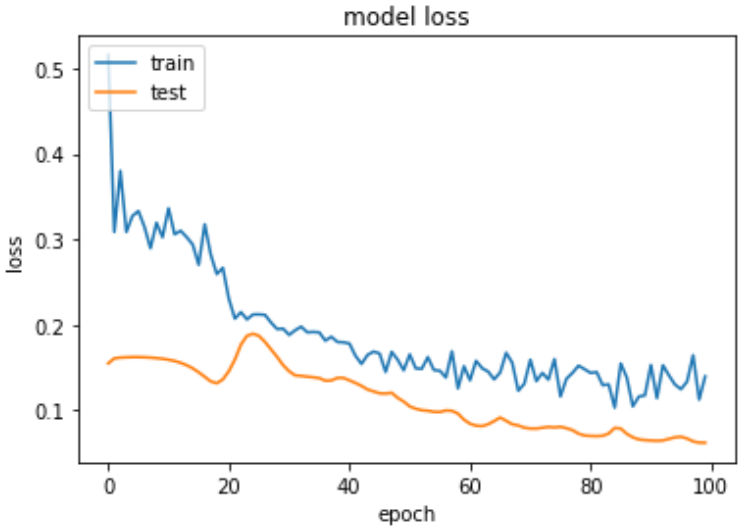

Figure 4.13: Model loss for CNN 5% Gene-selected Alzheimer's.

The final retrain was done with LSTM, which had the best results according to both the Cervical cancer and Alzheimer's disease datasets for percent 5 Gene-selected LSTM pieces of training, using a randomly chosen dataset. The opposite way round, the confusion matrix findings from LSTM seems to be completely satisfactory as well as the previous two model. The confusion matrix may be seen in Table 4.9.

Table 4.9: 5% Gene-selected Alzheimer LSTM Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	10	0
	Control	1	1

Similar to Deep Neural Network and CNN outcomes of Alzheimer's percent 5 gene selection for a randomly selected dataset, this retraining was 91.67 percent correct, and the model properly predicted the test values with only one mistake. Despite this, the recall was 1 and precision was 0.9, which was different from the previous models' retrain, yet the F1 score was 0.95 as same as the previous model. The accuracy and loss graphs are shown in Figures 4.15 and 4.16, respectively.

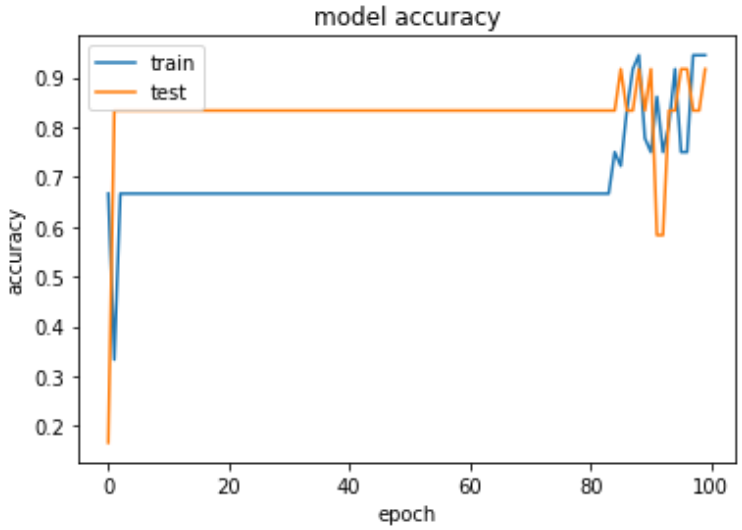

Figure 4.14: Model accuracy for LSTM 5% Gene-selected Alzheimer's.

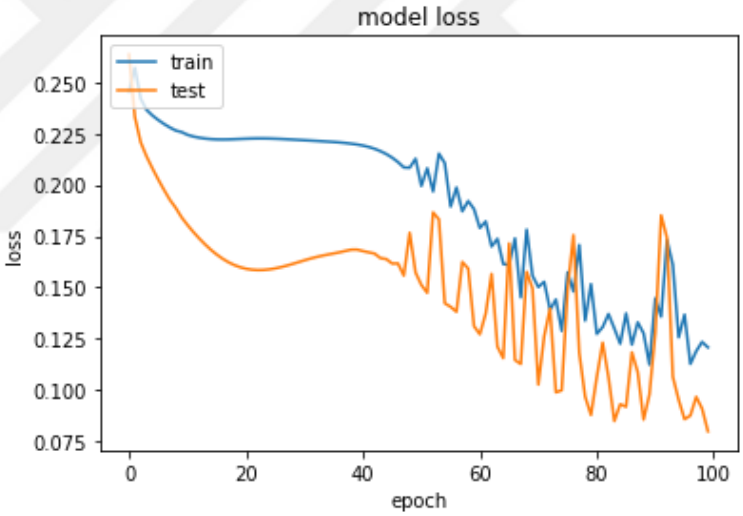

Figure 4.15: Model loss for LSTM 5% Gene-selected Alzheimer's.

When we looked at the training outcomes of two different datasets with 5% gene selection and the specified models, the order of success for cervical was D Deep Neural Network, CNN, and LSTM, while for Alzheimer's disease it was CNN, Deep Neural Network, and lastly LSTM. With two distinct hyperparameter values, the Deep Neural Network model produced the best performance (94.94%) in the Cervical dataset. Because it analyzes more samples and fewer but related genes, the deep learning model is more successful than the Alzheimer's data set (83.89%). As can be

seen, the CNN model and the Deep Neural Network model came quite close (92.12%). It also performed better than the Deep Neural Network result (83.89%) on the Alzheimer's set (84.74%). The difference between the two outcomes is relatively minor in this case. The rationale for this is that, due to the small number of samples in the Alzheimer's dataset, it appears that this sample size has formed a better relationship with 5% gene selection and a 1-dimensional convolutional layer, and has discovered the best value. However, when the number of genes grows, the convolutional layer may find this increasingly challenging. However, later gene selection experiments have revealed that this number has dropped. On the other hand, while the LSTM model yielded an 83.41% result in the cervical dataset and a lower result of 80.05% in the Alzheimer's dataset, this result appears respectable when compared to the prior two models' Alzheimer's outcomes. Furthermore, when compared to other models, these two outcomes are not favorable.

4.2.2 Case 2: 10% Gene Selection

The second step of training entails selecting the 10% of related genes. For the Cervical Cancer Dataset, 72 genes have been chosen. For the Alzheimer's data set, this value equals 281 Same as the first training, 50 datasets for Cervical cancer and 50 datasets for Alzheimer's disease were processed, with the selected genes being distinctive for each data collection. The training was processed after the selection was completed for both the Cervical and Alzheimer datasets. Same as before, there were 108 outcomes for each classification method and each disease in the first training results. After all of the training has been finished and the data compiled, the average of loss, mae, mape, accuracy, precision, recall, and f1 score values for each classification technique and disease are calculated.

Table 4.10 shows the average of the cervical cancer data sets' results for 10% gene-selected.

Table 4.10: Cervical Cancer dataset 10% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	**0.044**	**0.055**	**10.149**	**95.41%**	**0.991**	**0.924**	**0.954**
CNN	0.058	0.116	45.738	93.06%	0.969	0.899	0.929
LSTM	0.165	0.209	71.177	80.59%	0.876	0.750	0.782

For each classifier, Table 4.10 shows the most accurate (highest accuracy) findings. Table 4.10 also includes the findings for loss, mae, mape, accuracy, precision, recall, f1 score, and standard deviation. The most accurate model is DNN (Deep Neural Network), as seen in the table. The "Adam" optimizer, "0.005" learning rate, "75" epoch number, and "128" batch size are the most successful hyperparameters for Deep Neural Network results. As can be observed in Table 4.10, Deep Neural Network classifiers have the lowest error results, which is to be expected. Deep Neural Network model trained networks had the highest accuracy for 5% gene selection train case also.

However, the accurate ranking for cervical gene selection is the same as the percent 5 gene selection. Once again, the CNN classifier gives the closest accurate result. This classifier's hyperparameters are "Adamax" optimizer, "0.01" learning rate, "50" epoch number, and "32" batch size. With the hyperparameters "RMSProp" optimizer, "0.005" learning rate, "100" epoch number, and "32" batch size, the LSTM classifier has the lowest accuracy.

Apart from the Adadelta optimizer, three alternative optimizers, “Adam” for Deep Neural Network, “Adamax” for CNN, and “RMSProp” for LSTM, provided the greatest accuracy for their models and contributed to the best results at 10% gene selection. These optimizers required 128 and 32 batch sizes, as well as a 0.005 and 0.01 learning rate for 10% gene selection, to get the best results. The parameters connected with error calculation, such as Loss, Mae, and Mape, are lower when the accuracy value is bigger, as shown in the table, while the precision, recall, and F1 score values, which represent the validity of the result, are high in precise correlation to accuracy. To summarize, the error values for the maximum accuracy value are lower in the table, whilst the values representing the system's performance are greater.

The training was repeated using the hyperparameters where the classifier showed the best performance on a randomly selected data set to acquire the confusion matrices, accuracy, and loss graphs of these findings as it was done for 5% gene selection.

The Deep Neural Network model was used for the first retrain for the percent 10 gene-selected scenario, and it produced the best results. Cervical cancer datasets for 10% Gene-selected, as previously, using a randomly selected dataset. As a

consequence, the confusion matrix produced by Deep Neural Network is faultless; this confusion matrix is shown in Table 4.11.

Table 4.11: 10% Gene-selected Cervical Cancer Deep Neural Network Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	0
	Non-Tumor	0	9

Deep Neural Network was reassigned using the hyperparameters that deliver the best results: "Adam" optimizer, "0.005" learning rate, "75" epoch number, and "128" batch size. Figures 4.17 and 4.18 show the accuracy and loss graphs, correspondingly. The accuracy in this training was 100%; the system properly predicted the output of the test values without any faults. Similarly, the Recall, Precision, and F1 Score scores were all 1.

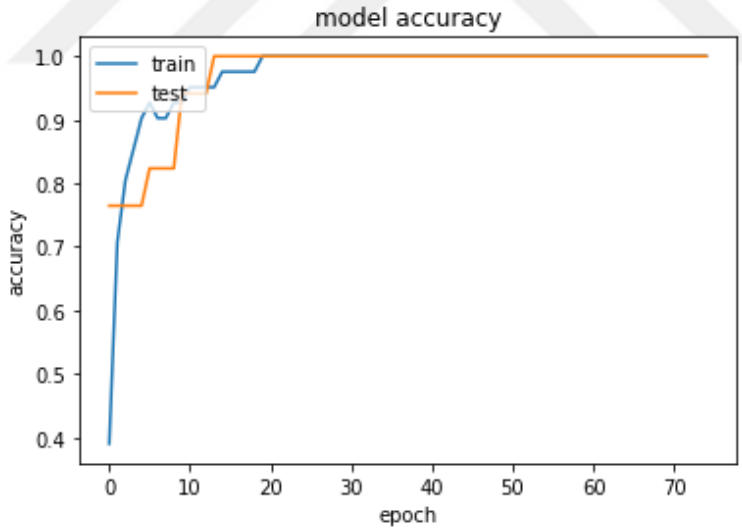

Figure 4.16: Model accuracy for Deep Neural Network 10% Gene-selected Cervical

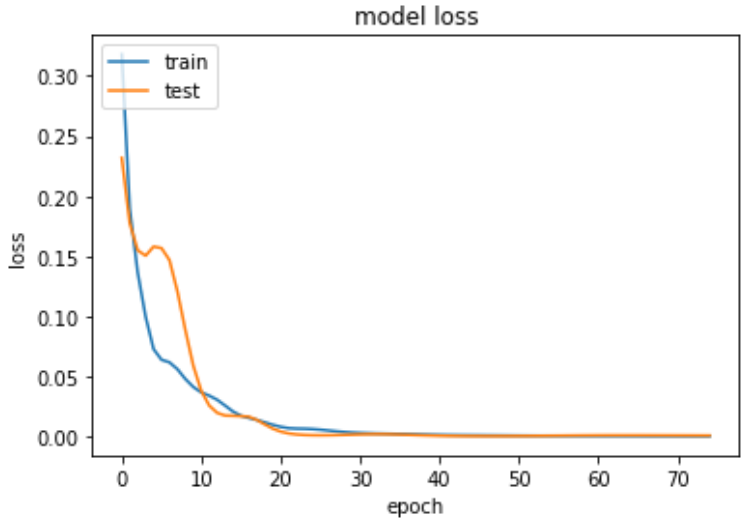

Figure 4.17: Model loss for Deep Neural Network 10% Gene-selected Cervical

For 10% Gene-selected Cervical cancer, the second iteration was evaluated using CNN, which yields the second-most accurate outcome. The confusion matrix of the CNN model returns the same results as the Deep Neural Network retrain confusion matrix. The confusion matrix is shown in Table 4.12.

Table 4.12: 10% Gene-selected Cervical Cancer CNN Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	0
	Non-Tumor	0	9

The following hyperparameters were used to perform CNN: "Adamax" optimizer, "0.01" learning rate, "50" epoch number, and "32" batch size. The training was 100 percent accurate, and the system correctly anticipated the output of the test values without any errors, much like the Deep Neural Network Confusion Matrix. The results for Recall, Precision, and F1 Score were all 1. The accuracy and loss graphs are shown in Figures 4.19 and 4.20.

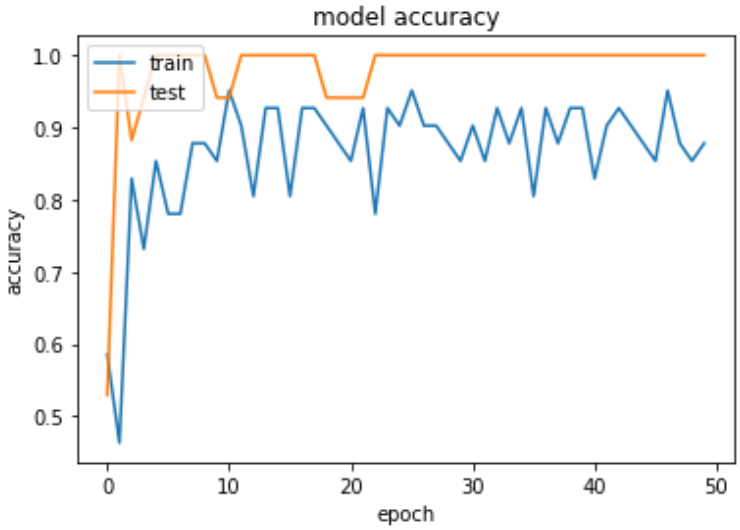

Figure 4.18: Model accuracy for CNN 10% Gene-selected Cervical

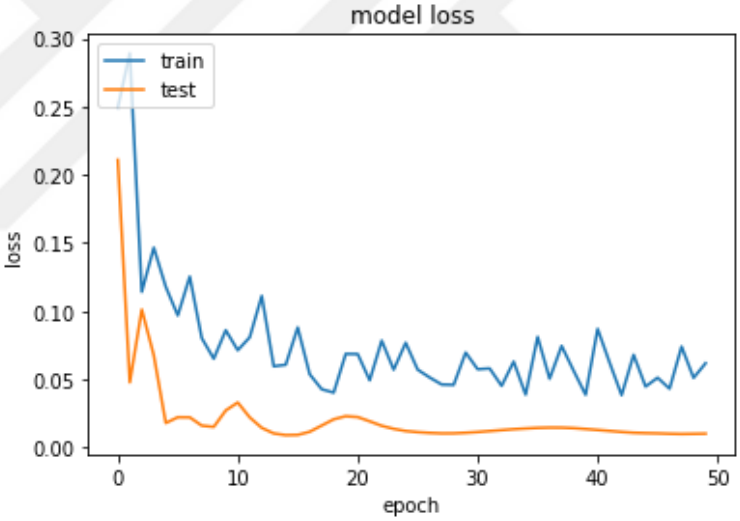

Figure 4.19: Model loss for CNN 10% Gene-selected Cervical

Finally, the LSTM retrain was performed, which had much lower accuracy than Deep Neural Network and CNN. The retraining is done by using identical hyperparameters on a randomly chosen dataset. In the other direction, the LSTM confusion matrix findings, as well as the preceding two models, appear to be satisfying. Table 4.13 displays the confusion matrix.

Table 4.13: 10% Gene-selected Cervical Cancer LSTM Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	0
	Non-Tumor	2	7

LSTM which had the lowest confusion matrix result was re-performed with the following hyperparameters: "RMSProp" optimizer, "0.005" learning rate, "100" epoch number, and "32" batch size. In contrast to the two prior systems, the training delivered the output of the test values with two errors. Even though the Precision value in the result was 1, the Recall was 0.82 and the F1 Score was 0.9. Figures 4.21 and 4.22 show the accuracy and loss graphs.

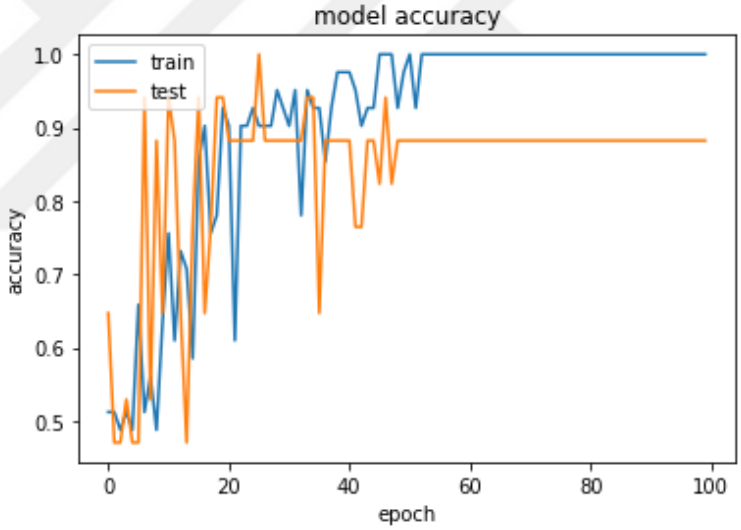

Figure 4.20: Model accuracy for LSTM 10% Gene-selected Cervical

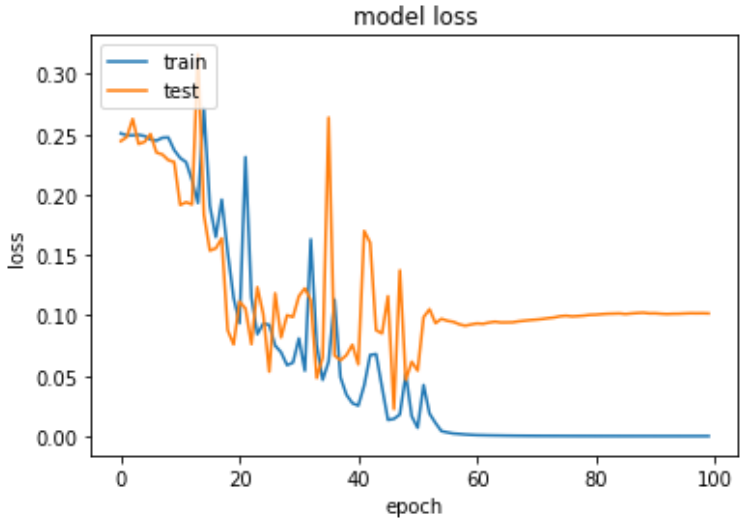

Figure 4.21: Model loss for LSTM 10% Gene-selected Cervical

After the Cervical Cancer training for the 10% gene-selected case was completed and the retraining results were acquired, Alzheimer's training was reconducted according to hyperparameters which returned the highest results. Table 4.14 shows the most accurate average of the Alzheimer dataset's results.

Table 4.14: Alzheimer dataset 10% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	**0.110**	**0.225**	**112.488**	**84.79%**	**0.890**	**0.910**	**0.891**
CNN	0.110	0.252	126.631	83.50%	0.870	0.928	0.887
LSTM	0.150	0.258	132.854	80.68%	0.865	0.872	0.857

Table 4.14, like Table 4.10, displays the most accurate (highest accuracy) findings as well as the loss, mae, mape, precision, recall, f1 score, and standard deviation values for each classifier.

For Alzheimer's disease, the alternative hyperparameter sets were more effective. The Deep Neural Network model is the best accurate classifier for Alzheimer's 10% gene selection, using the following hyperparameters for the results from the table: "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "64" batch size. The most accurate hyperparameters for Alzheimer's disease 5% gene

selection were "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "128" batch size, all of the hyperparameters are identical except batch size with the most correct hyperparameter in this system. As demonstrated in table 4.12, the CNN classifier has the same accuracy as the cervical cancer percent 10 gene selection. " Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "64" batch size were the hyperparameters for this classifier. The LSTM classifier has the lowest accuracy in Alzheimer's 10% gene selection data sets using the hyperparameters "Adam" optimizer, "0.005" learning rate, "100" epoch number, and "64" batch size. If we compare LSTM hyperparameters of Alzheimer's 5% gene selection and this system, we can examine that almost all the parameters are the same as the Deep Neural Network comparison.

The Adamax optimizer, according to the metrics, produced the best results for the Alzheimer data set at 10% gene selection, just as it did for the data set with 5% gene selection. To obtain the best results with this optimizer for this dataset, 64 batch sizes and a 0.01 learning rate were recommended. The system likewise performs better with the higher epoch number. As previously stated in the preceding section, the parameters linked with error calculation, such as Loss, Mae, and MAPE, drop as the accuracy value increases; nevertheless, precision, recall, and F1 score values show the validity of the result, increase in direct proportion to accuracy. The error numbers in the table that indicate the highest level of accuracy are lower, while the values that reflect the system's performance are higher, as before.

The training was repeated using the hyperparameters where the classifier performed best on a randomly chosen data set to get the confusion matrices, accuracy, and loss graphs of these findings, much as it was done for the cervical cancer dataset before.

The initial retrain was tested using a Deep Neural Network for 10% Gene-selected Alzheimer's disease. The Deep Neural Network model's confusion matrix shows that the retrain is 100% accurate. Table 4.15 displays the confusion matrix.

Table 4.15: 10% Gene-selected Alzheimer Deep Neural Network Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	9	0
	Control	0	3

The following hyperparameters were used to perform Deep Neural Network: "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "64" batch size. As can be seen in the accuracy and loss graphs which are shown in Figures 4.23 and 4.24, respectively, the algorithm correctly anticipated the outcome of the test values without any error. The Recall, Precision, and F1 Score ratings were all 1.

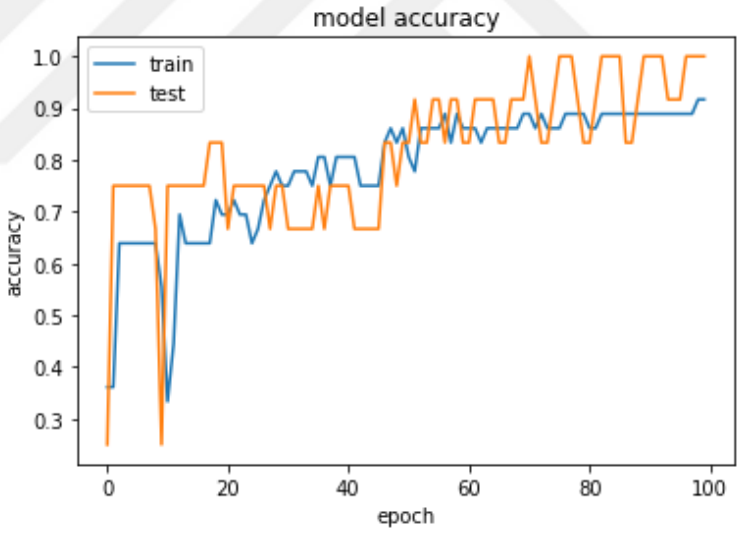

Figure 4.22: Model accuracy for Deep Neural Network 10% Gene-selected Alzheimer's.

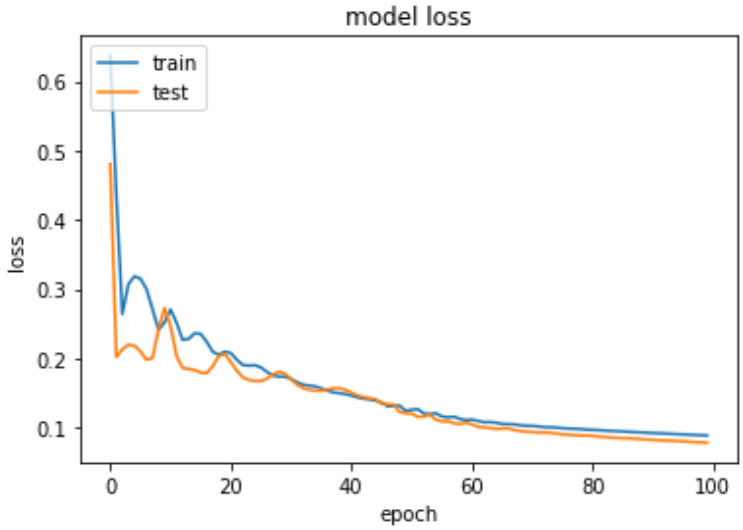

Figure 4.23: Model loss for Deep Neural Network 10% Gene-selected Alzheimer's.

For 10% Gene chosen Alzheimer's disease, the second retrain was evaluated using the CNN model, which delivers the second-most accurate outcome. The confusion matrix of the CNN model provides the same findings as the Deep Neural Network retrain confusion matrix, indicating that there was no mistake in the CNN retrain. The confusion matrix is shown in Table 4.16.

Table 4.16: 10% Gene-selected Alzheimer CNN Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	9	0
	Control	0	3

The retrain of CNN was processed with the hyperparameter set that is "Adamax" optimizer, "0.01" learning rate, "75" epoch number, and "64" batch size. The hypermeter set is almost the same as the Deep Neural Network Alzheimer 10% gene selection hyperparameter setting. The algorithm properly predicted the outcome of the test values without any mistakes as same as the previous retrain, as indicated in

the accuracy and loss graphs in Figures 4.25 and 4.26, respectively. The scores for Recall, Precision, and F1 Score were all 1.

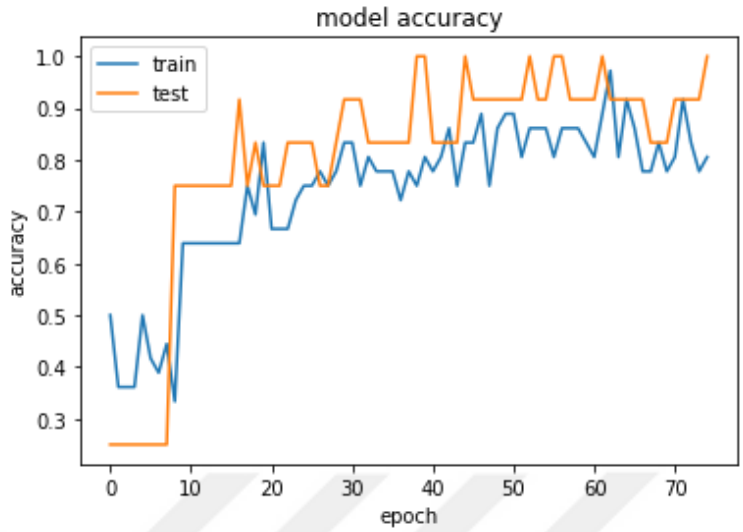

Figure 4.24: Model accuracy for CNN 10% Gene-selected Alzheimer's.

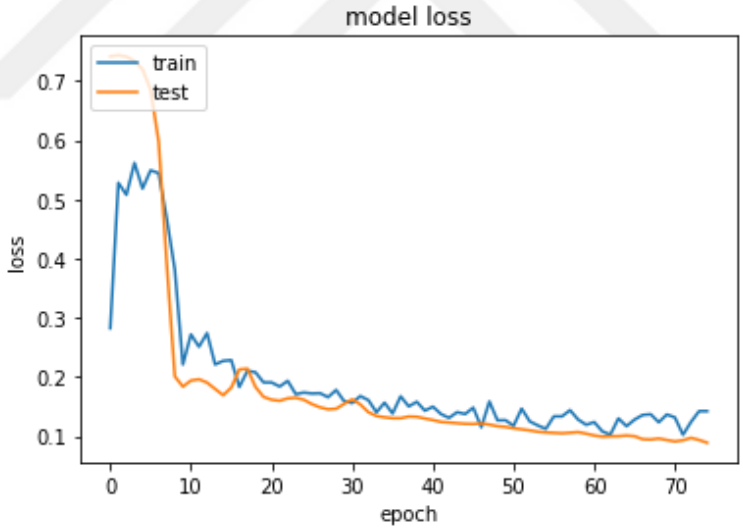

Figure 4.25: Model loss for CNN 10% Gene-selected Alzheimer's.

Using a randomly selected dataset, the final retrain was done with LSTM, which is approximately as accurate as cervical cancer 10% gene selection LSTM model. Aside from that, despite its poor accuracy, the confusion matrix findings from LSTM appear to be sufficient. Table 4.17 depicts the confusion matrix.

Table 4.17: 10% Gene-selected Alzheimer LSTM Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	8	1
	Control	1	2

LSTM confusion matrix demonstrates that there were two errors when the system was evaluated for Alzheimer's 10% gene selection for a randomly selected dataset, notwithstanding the prior two outcomes. Hence none of the parameters are 1, precision, recall, and f1 score are all 0.89. Figures 4.27 and 4.28 show the accuracy and loss plots, respectively.

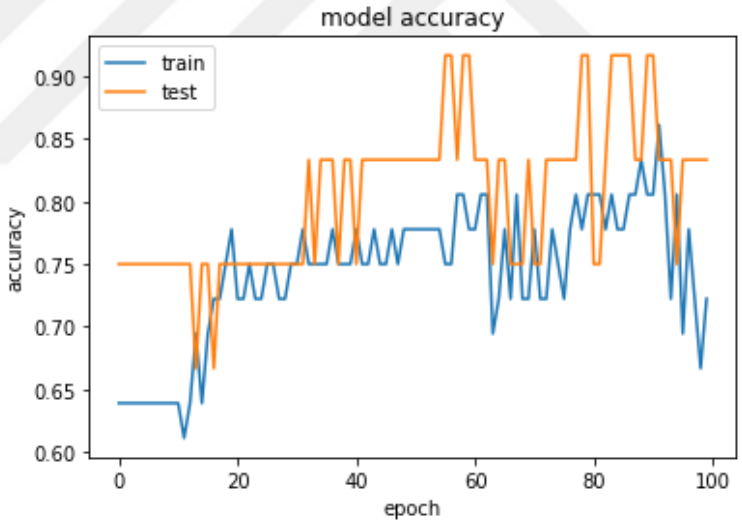

Figure 4.26: Model accuracy for LSTM 10% Gene-selected Alzheimer's.

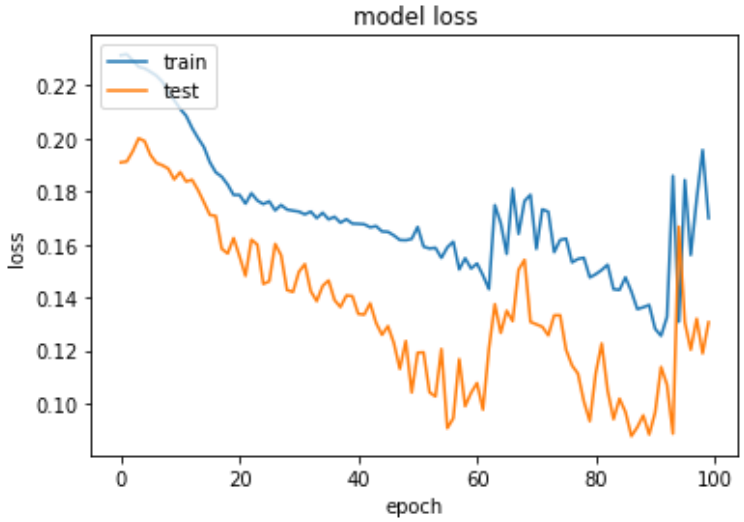

Figure 4.27: Model loss for LSTM 10% Gene-selected Alzheimer's.

The orders of success for cervical cancer and Alzheimer's disease were the same when we looked at the training outcomes of two distinct datasets with 10% gene selection and the specified models: Deep Neural Network, CNN, and LSTM. The Deep Neural Network model generated the greatest performance (95.41%) in the Cervical dataset, which is greater than cervical cancer 5% gene selection best result (94.94%) acquired by Deep Neural Network. This may be due to the achievement of optimum gene selection. The training for 10% gene selection in the Cervical Cancer data set is greater than the training for 10% gene selection in the Alzheimer data set, as it was for 5% gene selection. The argument for the Alzheimer's samples is not as compelling as the reason for the cervical cancer sample to link genes and samples. The system examines fewer samples with more genes for Alzheimer's, and it is lowering the system's robustness. The Deep Neural Network model performed worse on the prediction of the Alzheimer dataset, with accuracy dropping from 95.41 percent to 84.79 percent. The CNN model is the second most successful model for both diseases, but with a notable difference: the accuracy of the CNN model for cervical cancer is 93.06 percent, but it is only 83.5 percent for the Alzheimer's data set. As can be observed, in this case, the CNN and Deep Neural Network models got quite near to the prior one. In this situation, the difference between the two results is minimal. However, when the number of genes rises, the convolutional layer may find this more difficult to relate to Alzheimer's samples, resulting in a fall in CNN accuracy from 84.74% to 83.50%. On the other hand, for Cervical samples, the system worked robust with more genes, and the accuracy result increased from 92.12% to 93.06%. The

LSTM model, on the other hand, produces nearly identical findings for both illnesses' 10% gene selection situation. The accuracy of the cervical data set is 80.59%; however, the accuracy of the Alzheimer data set has grown (80.68%). Even though accuracy is marginally improved, two LSTM outputs are unacceptable when compared to other models.

4.2.3 Case 3: 30% Gene Selection

The final stage of training is to choose the top 30% of associated genes. 215 genes were evaluated for the Cervical Cancer Dataset. This quantity is 841 in the Alzheimer's sample group. 50 datasets for Cervical cancer and 50 datasets for Alzheimer's disease were analyzed in the same style as the prior training, with the indicated genes being unique for each data collection. Both the Cervical and Alzheimer datasets had their training run once the sampling was done. In the last two training outcomes, there were 108 outputs for each classification technique and each condition, the same as before. The average of loss, mae, mape, accuracy, precision, recall, and f1 score values for each classification technique and condition are determined once all of the training is completed and the data is obtained.

The average of the cervical cancer data sets' results for the percent 10 gene chosen is shown in Table 4.18.

Table 4.18: Cervical Cancer dataset 30% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	**0.046**	**0.061**	**21.221**	**94.94%**	**0.969**	**0.937**	**0.950**
CNN	0.052	0.107	51.947	93.18%	0.951	0.923	0.931
LSTM	0.264	0.321	88.957	70.24%	0.712	0.567	0.590

Table 4.18 displays the most accurate (highest accuracy) results for each classifier’s average value of 50 cervical dataset’s results. The results for loss, mae, mape, accuracy, precision, recall, f1 score, and standard deviation are also shown in Table 4.18. As seen in the table, the most reliable model is DNN (Deep Neural Network). The most effective hyperparameters for Deep Neural Network outcomes are the "Adam" optimizer, "0.005" learning rate, "75" epoch number, and "32" batch

size. These hyperparameters are nearly identical to the hyperparameter set for the Deep Neural Network 10% Gene-selected Cervical Dataset. Deep Neural Network classifiers had the lowest error outcomes, as seen in Table 4.18. Also, for the 5% and 10% gene selection train cases, Deep Neural Network model-generated networks had the greatest accuracy. Cervical Dataset 30% gene selection result has the same accurate ranking as 5% and 10% gene selection. Nonetheless, when compared to the 10% gene selection Cervical Cancer result, Deep Neural Network's accuracy is lower. The CNN classifier provides the second most accurate result once again. The "Adamax" optimizer, "0.01" learning rate, "75" epoch number, and "128" batch size are the hyperparameters for this classifier. The CNN model likewise produced the model's own second greatest result with nearly identical hyperparameters to the CNN model's hyperparameters and offered the best result for 10% gene selection. According to previous findings, the LSTM classifier has the lowest accuracy when using the hyperparameters "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "32" batch size as the prior results.

At 30% gene selection, "Adam" with CNN and Deep Neural Network and "Adamax" with LSTM offered the highest accuracy for their models and contributed to the highest performance. To get the best outcome, these optimizers used 32 and 128 batch sizes, as well as a 0.005 and 0.01 learning rate for 30% gene selection. As indicated in the table, the parameters related to error computation, such as Loss, Mae, and Mape, are lower as the accuracy value increases, but precision, recall, and F1 score values, which represent the result's validity, are high in exact correlation to accuracy as it explained previous sections.

As was done for percent 5 and percent 10 gene selection, training was repeated using the hyperparameters where the classifier performed best on a randomly selected data set to get the confusion matrices, accuracy, and loss graphs of these discoveries.

For the initial retrain for the percent 30 gene chosen scenario, the Deep Neural Network model was utilized, and it delivered the best results. As before, a randomly selected dataset was used to create cervical cancer datasets for the 30% gene-selected. As a result, the confusion matrix generated by Deep Neural Network indicates that the retrain is not as successful as prior Cervical retrains (see Table 4.19).

Table 4.19: 30% Gene-selected Cervical Cancer Deep Neural Network Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	6	3
	Non-Tumor	0	8

The hyperparameter set which is the "Adam" optimizer, "0.005" learning rate, "75" epoch number, and "32" batch size were used to retrain Deep Neural Network. The accuracy and loss graphs are shown in Figures 4.29 and 4.30, respectively. The findings reveal that for percent 30 gene selection, the accuracy is lower and the model's loss values are higher. The reason for this may be due to a randomly selected dataset, which means the correlation between samples and genes can be imprecise; if the classifier can't relate samples with genes that are nearly three times as many as in the percent 10 gene selection case, the result will not be as successful as possible.

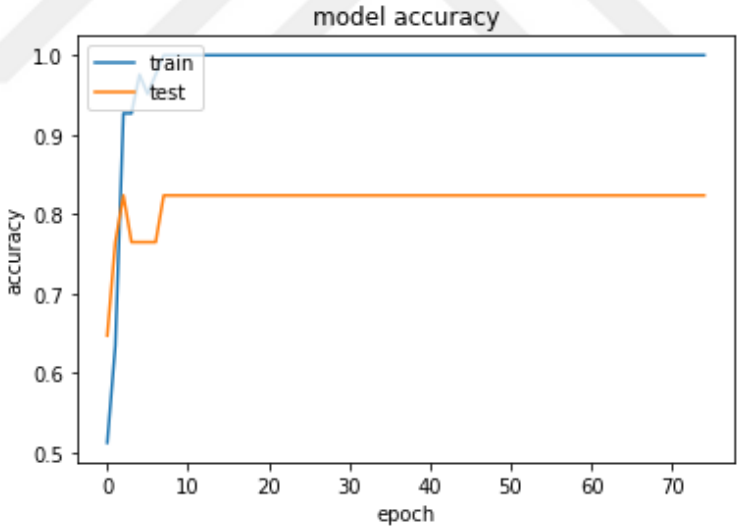

Figure 4.28: Model accuracy for Deep Neural Network 30% Gene-selected Cervical

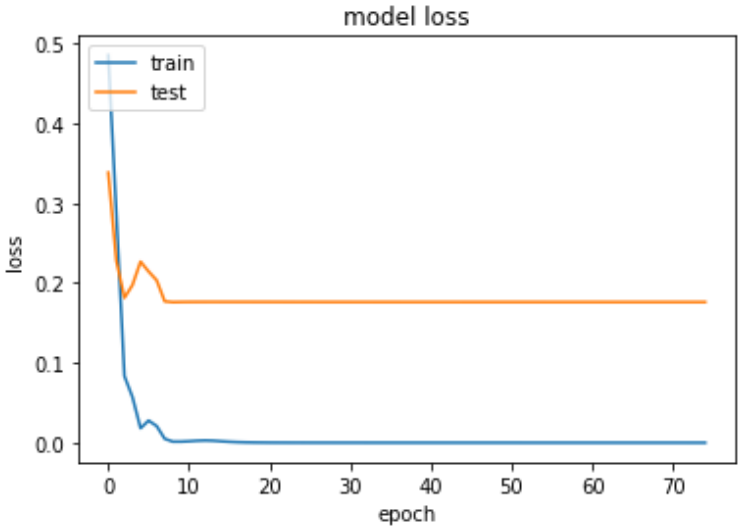

Figure 4.29: Model loss for Deep Neural Network 30% Gene-selected Cervical

The second iteration was analyzed using CNN, which produces the second-most accurate average result for the percent 30 Gene-selected Cervical cancer. The CNN model's confusion matrix produces a similar performance to the Deep Neural Network retrains confusion matrix. Table 4.20 shows the confusion matrix.

Table 4.20: 10% Gene-selected Cervical Cancer CNN Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	7	2
	Non-Tumor	1	7

CNN was performed with the following hyperparameters: "Adamax" optimizer, "0.01" learning rate, "75" epoch number, and "128" batch size. The training was 73.7 percent accurate, and the system properly predicted the output of the test values with only three mistakes, which was nearly identical to the Deep Neural Network Confusion Matrix. The results are 0.78 for Recall, 0.89 for Precision, and 0.83 for F1 Score. Figures 4.31 and 4.29 show the accuracy and loss graphs.

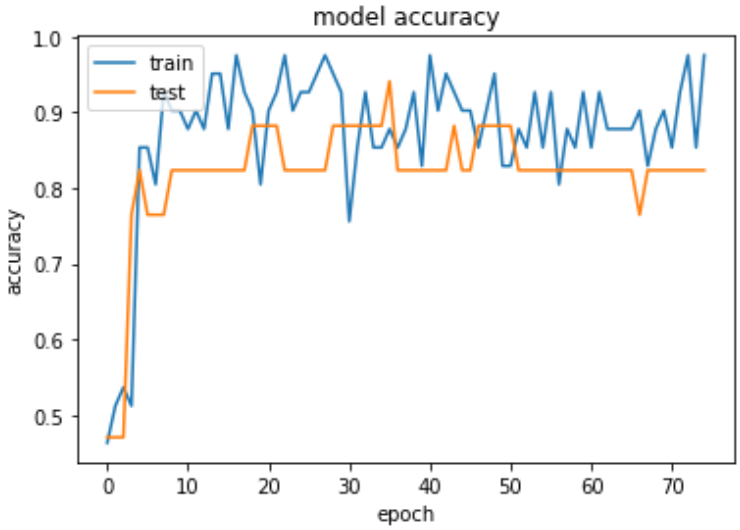

Figure 4.30: Model accuracy for CNN 30% Gene-selected Cervical

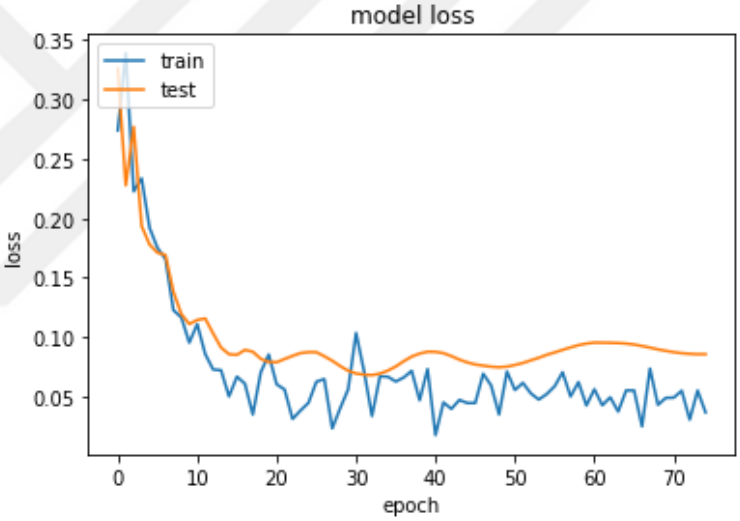

Figure 4.31: Model loss for CNN 30% Gene-selected Cervical

The most recent retrain was for LSTM, which fared marginally better than Deep Neural Network and CNN in terms of accuracy. The retrain is carried out with the same hyperparameters on a dataset selected at random. As a result, when comparing the two models, the LSTM confusion matrix outputs appear to be more successful. The confusion matrix is shown in Table 4.21.

Table 4.21: 30% Gene-selected Cervical Cancer LSTM Confusion Matrix

		Predicted Values	
		Tumor	Non-Tumor
Actual Values	Tumor	8	1
	Non-Tumor	1	7

The LSTM that produced the highest confusion matrix for retaining was re-run using the following hyperparameters: "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "32" batch size. This model uses the same optimizer and learning rate as CNN percent 30 gene selection. LSTM has the lowest accuracy rate for average value when compared to the other two classifiers, however, the training output of the test values is larger than the retrain accuracy result. There are two errors in the retrain result. The Precision and Recall numbers are both 0.89, and the same value for the F1 Score. The accuracy and loss graphs are shown in Figures 4.33 and 4.34.

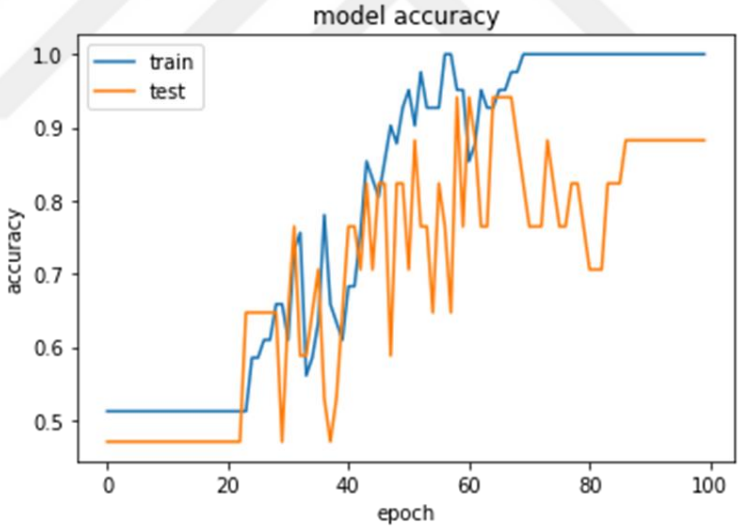

Figure 4.32: Model accuracy for LSTM 30% Gene-selected Cervical

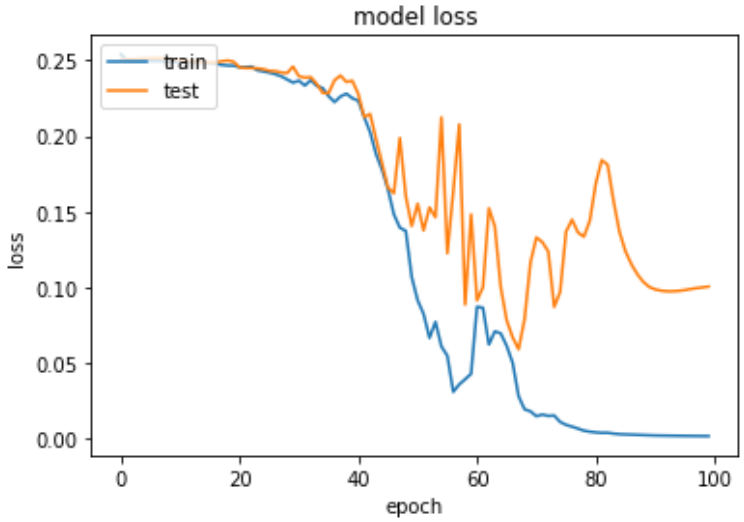

Figure 4.33: Model loss for LSTM 30% Gene-selected Cervical

Following the completion of Cervical Cancer training for the percent of 30 gene-selected cases and the acquisition of retraining results, Alzheimer's training was retrained using hyperparameters that yielded the best average outcomes. The most accurate average of the Alzheimer's dataset's results is shown in Table 4.22.

Table 4.22: Alzheimer dataset 30% gene selection results

Classifier	Loss	Mae	Mape	Accuracy	Precision	Recall	F1
DL	**0.124**	**0.242**	**121.963**	**84.56%**	**0.872**	**0.938**	**0.895**
CNN	0.107	0.234	127.464	83.53%	0.849	0.954	0.890
LSTM	0.189	0.371	175.300	72.09%	0.773	0.910	0.815

The greatest accuracy from the combination of hyperparameters findings for an average of 50 datasets results is shown in Table 4.22, together with average values of the loss, mae, mape, precision, recall, f1 score, and standard deviation for each classifier.

The almost same hyperparameter settings were utilized for Alzheimer's disease, and models found this hyperparameter to be more beneficial. "Adamax" was chosen as the optimizer in all three models, with "0.01" as the learning rate and 100 epochs. The batch sizes for Deep Neural Network and CNN are both 128, whereas

LSTM utilizes a 64-batch size. For Alzheimer's percent 30 gene selection, the Deep Neural Network model is the most accurate classifier. Even though, the accuracy is decreased when we compare it to the Cervical 30% gene-selected Deep Neural Network accuracy result. Same as Deep Neural Network, CNN accuracy is decreased as well from 93.18% to 83.53%, although the CNN model outcomes the second-best accuracy for the 30% gene-selected Alzheimer data set. The LSTM classifier has the lowest accuracy in Alzheimer's 30% gene selection data sets.

Adamax optimizer delivered the best results for the Alzheimer data set at 30% gene selection, just as it did for the data set with 5% and 10% gene selection. For this dataset, 128 batch sizes and a 0.01 learning rate were advised for the best results with this optimizer. The greater the epoch number, in this example 100, the better the system operates. As previously stated in the preceding section, as the accuracy value increases, the parameters associated with error calculation, such as Loss, Mae, and MAPE, decrease; however, precision, recall, and F1 score values, which demonstrate the validity of the result, increase in direct proportion to accuracy.

The training was repeated using the hyperparameters where the classifier performed best on a randomly chosen data set to get the confusion matrices, accuracy, and loss graphs of these findings, similar to how it was done before for the cervical cancer dataset.

For 30% of Gene-selected Alzheimer's disease, the initial retrain was evaluated using a Deep Neural Network. The result is almost perfect, the system fails for only one output. Table 4.23 shows the confusion Matrix of 30% gene-selected Alzheimer's Deep Neural Network model results.

Table 4.23: 30% Gene-selected Alzheimer Deep Neural Network Confusion Matrix

		Predicted Values	
		Alzheimer’s Diseased	Control
Actual Values	Alzheimer’s Diseased	11	0
	Control	1	0

As can be seen in the accuracy and loss graphs which are shown in Figures 4.35 and 4.36, respectively, the algorithm correctly anticipated the outcome of the test values with only one error. The Recall is 1, Precision on the other hand is 0.92 and the F1 Score value is 0.94. The randomly selected data set has only one negative which is the control sample in the dataset, and the system may have some difficulty correlating this sample to over 800 genes.

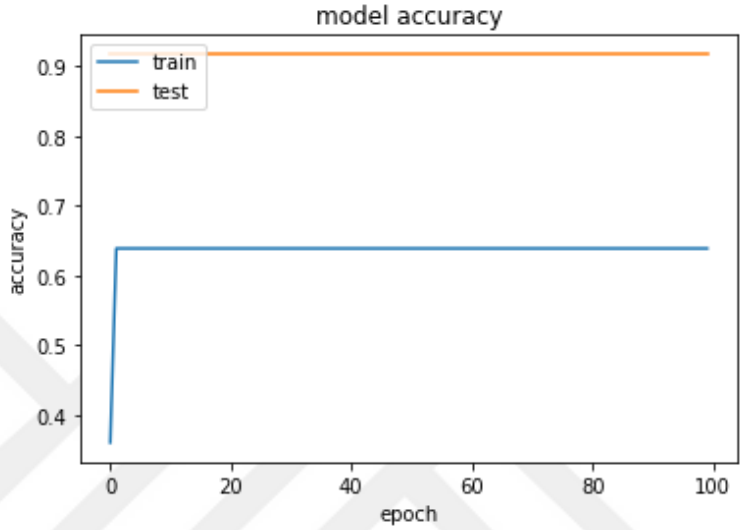

Figure 4.34: Model Accuracy for Deep Neural Network 30% Gene-selected Alzheimer

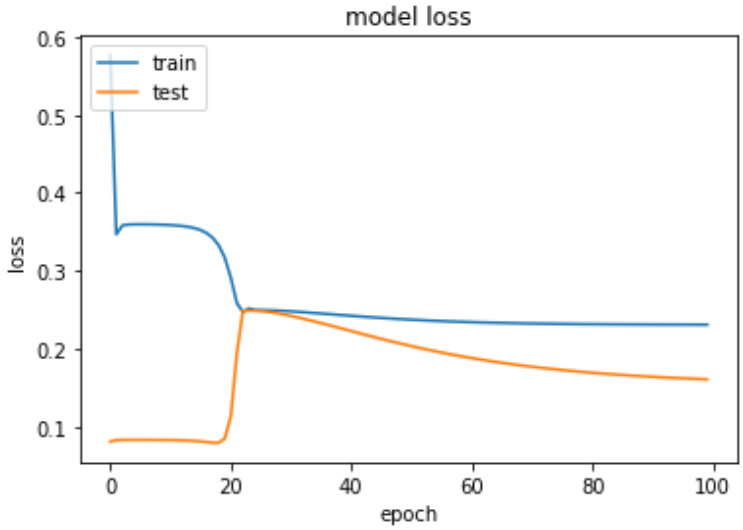

Figure 4.35: Model Loss for Deep Neural Network 30% Gene-selected Alzheimer

The second retrain was tested using the CNN model for the percent 30 Gene-selected Alzheimer's disease, which offers the second most accurate output in terms of average accuracies. The CNN model's confusion matrix yields nearly identical results to the Deep Neural Network retrain confusion matrix, showing that the CNN retrain only made one mistake. The algorithm fails to predict one sick sample in this example. Table 4.24 shows the confusion matrix.

Table 4.24: 30% Gene-selected Alzheimer CNN Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	10	1
	Control	0	1

The hypermeter set used for the CNN retrain was "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "128" batch size. The CNN Alzheimer's percent 10 gene selection hyperparameter set is nearly identical to the hyperparameter set. The accuracy and loss graphs in Figures 4.37 and 4.38, respectively, show that the algorithm correctly anticipated the outcome of the test value. Precision is 1 in this scenario, while Recall is 0.92, and the F1 Score is 0.94 once more.

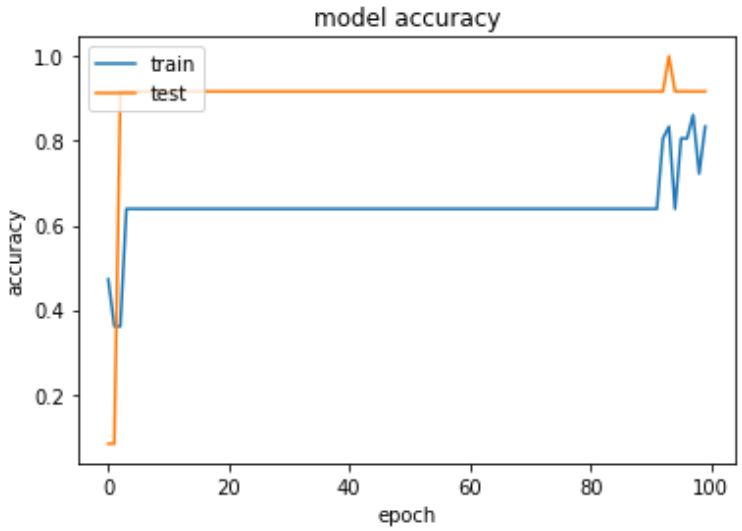

Figure 4.36: Model Accuracy for CNN 30% Gene-selected Alzheimer

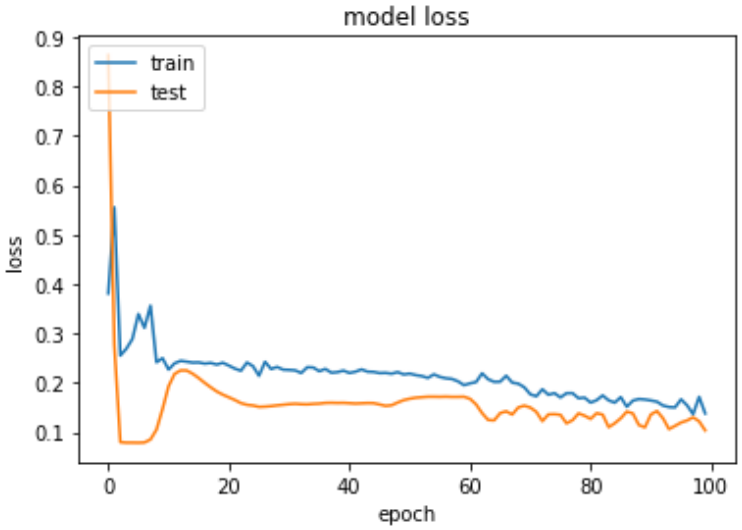

Figure 4.37: Model Loss for CNN 30% Gene-selected Alzheimer

The final retrain was done with LSTM, which is about as accurate as the cervical cancer percent 30 gene selection LSTM model, using a randomly selected dataset. Aside from that, the LSTM confusion matrix findings appear to be great, despite their poor accuracy. The confusion matrix is shown in Table 4.25.

Table 4.25: 30% Gene-selected Alzheimer LSTM Confusion Matrix

		Predicted Values	
		Alzheimer's Diseased	Control
Actual Values	Alzheimer's Diseased	10	1
	Control	0	1

The retrain results are the same as the CNN retrain of the Alzheimer dataset with a percent 30 gene selection. The "Adamax" optimizer, "0.01" learning rate, "100" epoch number, and "64" batch size were utilized in the LSTM retrain. The accuracy and loss graphs in Figures 4.39 and 4.40, respectively, demonstrate that the algorithm accurately predicted the test value's outcome. Precision is 1, Recall is 0.92, and F1 Score is 0.94, which is the same as the previous retrain.

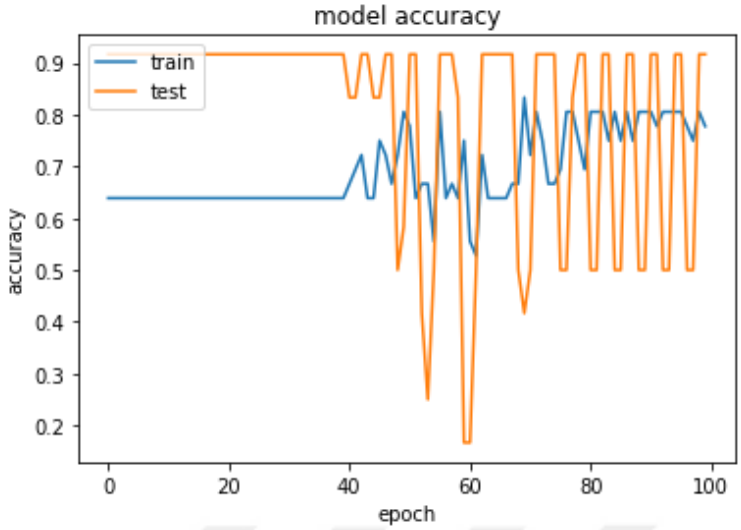

Figure 4.38: Model Loss for LSTM 30% Gene-selected Alzheimer

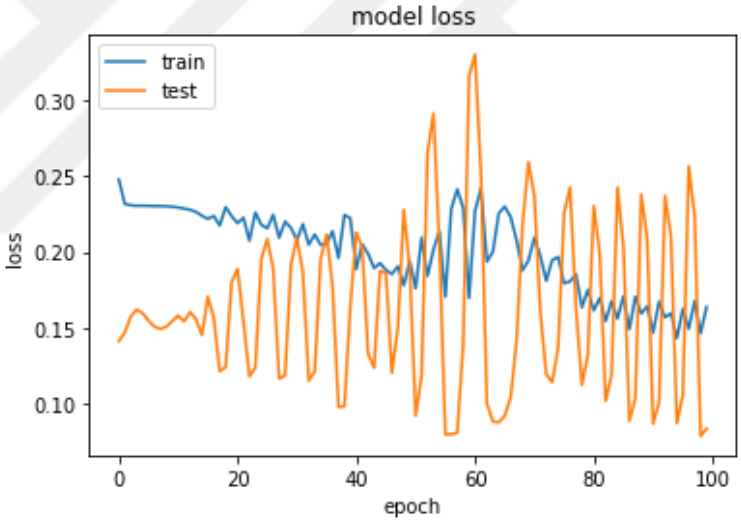

Figure 4.39: Model Loss for LSTM 30% Gene-selected Alzheimer

When it comes to examining the data, with the training outputs of two independent datasets with 30% gene selection and the stated models: Deep Neural Network, CNN, and LSTM, the orders of success for cervical cancer and Alzheimer's disease were the same. In the Cervical dataset, the Deep Neural Network model had the highest performance (94.94%), which is the same as the cervical cancer percent 5 gene selection best result obtained by Deep Neural Network. Furthermore, when compared to the best result obtained by Deep Neural Network for cervical cancer

percent 10 gene selection (95.41%), the best outcomes of this scenario are lower. The Cervical Cancer data set's training for 30% gene selection is higher than the training for 30% gene selection in the Alzheimer data set, as it was for 5% and 10%. On the Alzheimer dataset, the Deep Neural Network model fared worse, with accuracy dropping from 94.94% to 84.56%. The CNN model is the second most successful model for both conditions, but there is a significant difference: the CNN model's accuracy for cervical cancer is 93.18%, but only 83.53% for the Alzheimer's data set. As can be seen, the CNN and Deep Neural Network models in this example came quite close to the previous one. In this case, the difference between the two outcomes is negligible. When the number of genes increases, however, the convolutional layer behaves similarly to how it behaved during cervical cancer training, resulting in a modest increase in CNN accuracy from 83.50 to 83.53 for the Alzheimer data set for this gene selection. At the same time, the accuracy of cervical cancer screening improves as the number of genes grows. The LSTM model, on the other hand, generates pretty much identical results for the percent 30 gene selection condition in both disorders. The accuracy of the cervical data set is 70.24 percent; however, the accuracy of the Alzheimer data set has improved (percent 72.09), as it did for the percent 10 gene-selection. Even if Alzheimer's accuracy is marginally improved, two LSTM outputs are poor when compared to other models, and their accuracy drops from 80% to 70%.

4.3. Discussion

As shown in the Result section, data sets for two different health problems were trained and their performances were tested with 3 classification methods according to 5%, 10%, and 30% gene selections.

According to the results of these training and the test, the highest accuracy result was revealed with 95.41% results by training the Cervical Cancer dataset with the Deep Neural Network model in the 10% gene selection system. This result was higher than the 94.4% accuracy result obtained by using the voomNSC, sNBLDA, and NBLDA models for the cervical cancer dataset in the study of Zararsız et al. At the same time, the Deep Neural Network model, with an accuracy of 94.94% in 5% and 30% gene selection, slightly surpassed the 94.4% result achieved by Goksuluk et al (2019). In this study, the Support Vector Machine, which is also a traditional classification method, was used for Cervical cancer classification and got an accuracy

of 88.9%. With the success of the Deep Neural Network, it is obvious that Goksuluk gave more successful results than the interface called MLSeq, which was put forward by the models that others found as a result of this study. On the other hand, the other two models could not achieve the 94.4% accuracy success achieved by Gökmen and others for Cervical Cancer. The most successful Cervical Cancer rate for CNN was obtained in 30% gene selection, 93.18%, which is very close to the result of Goksuluk et al (2019). The results of the CNN model for Cervical Cancer are also not very different from these results; It is 92.12% for 5% gene selection and 93.06% for 10% gene selection. However, the LSTM model has not been as successful as these two models for Cervical cancer. The highest accuracy rate was 83.41% and was obtained in 5% gene selection. As the rate of gene selection increased, so did the LSTM accuracy rate; While the result with 10% gene selection was 80.59%, for 30% it was considerably reduced and accuracy gave a result of 70.24%.

In the training and testing of the Alzheimer's disease set, the success achieved in the Cervical Cancer set could not be achieved. Deep Neural Network's model revealed the highest success result for Alzheimer's with 84.79% in 10% gene selection. A similar study to this study was carried out by Zararsız G. in his doctoral thesis presented in 2019. Two studies related to this thesis have been presented, and according to the results here, the highest accuracy of Alzheimer's classification is 78.9% with the NSC2 model (Goksuluk D., 2019). In the studies related to the thesis, the accuracy rate was not presented as a number, instead, the error rate was shared. While this value was 8.7% and 13.9% in the SVM and voomDQDA algorithms model, the test errors were quite high with a rate of 31.7% (Zararsız G et al., 2019). In the same study, the SVM model was calculated with the lowest misclassification error rate of 0.087 for the shared data. In this study, the minimum loss value measured by the MSE method in the Alzheimer's set is 0.107. It was obtained in the 30% gene selection of the CNN model. In the studies conducted by Goksuluk, 70 sample sets, 48 Alzheimer's and 22 Control samples were used, while 49 samples were used in this study. While this was one of the most important reasons for the difference, it was hopeful that the results were close. While the success order for Alzheimer's was CNN, Deep Neural Network, and LSTM for 5% gene selection, Deep Neural Network showed more successful results for 10% and 30% gene selection, and LSTM gave low results in the system.

Examining the performances of the most successful results helps us see that the hyperparameters they provide are close to each other. At the same time, we can see that similar hyperparameter sets are used for the same disease. For example, Deep Neural Network training, which is the most successful Model for the Cervical set, was provided with the Adam optimizer and with a learning rate of 0.005 for this optimizer. Deep Neural Network obtained the best four results (two results are added to the table since the accuracy in 5% gene selection is the same: see table 4.2) with 32 and 128 batch sizes and it provided them with an epoch value of 75. Similarly, Deep Neural Network, which is the most successful model in 10% and 30% gene selection for Alzheimer's set (it gave results very close to the most successful system in 5% gene selection), used the Adamax optimizer with a learning rate of 0.01 for these three scenarios. All three systems had an epoch count of 100 and kept their batch sizes high overall, 128,64,128 respectively. Similarly, the CNN model also gave the most successful results when training the same optimizer with the same learning rate for Alzheimer's datasets, while this optimizer was Adamax, the learning rate was 0.01. In addition, for Alzheimer's, CNN mostly preferred 100 epochs and 128 batch sizes. On the other hand, CNN Cervical has not always used the same optimizer on its set. CNN, which is the second most successful system for this disease, preferred Adamax in 10% and 30% gene selection, while it preferred Adam optimizer in the 5% gene selection scenario. While Adamax used a 0.01 learning rate with his optimizers, he used a 0.001 learning rate with Adam, the only result using this learning rate among all the best results. Besides, he mostly preferred 50 and 32 values for epoch and batch sizes. Apart from these, the LSTM model also preferred different hyperparameter sets, similar to the CNN model. While choosing LSTM Adamax and RMSProp optimizers for Cervical training, they used these optimizers with a learning rate of 0.01 and 0.005. The epoch number was 100 for the three best results, while the batch sizes were 32 and 128. It is similar to the Alzheimer's disease set, where his model outperforms Cervical Cancer in 10% and 30% gene selections. These training and tests used the LSTM model, which preferred Adam and Adamax optimizers, and used 0.005 and 0.01 learning rates in the optimizers. As in the previous scenario, it was more successful in this system with 100 epochs. Batch sizes have been chosen as 64 and 128.

When we examine the results in more detail, while the optimal gene selection for Deep Neural Network was 10%, the system was more successful when the gene selection for CNN increased, so 30% gene selection was the most successful scenario. Unlike these two, LSTM gave its best results in the 5% scenario with the lowest gene selection.

In the Deep Neural Network model, we see that the relationship between genes and samples is best established in 10% of gene selection. This situation is the same for the results of both diseases. This uncomplicated system with a smaller number of layers has successfully established relationships at lower and higher rates and has been the most robust system. The Deep Neural Network generally gives promising results for data classification Seq Data as it did in the system (Rukhsar, L. et al., 2022).

The reason why the CNN system is not successful in Deep Neural networks can be attributed to the mistakes it may have made in the further calculations it has made in the layers it uses to associate too many features with samples. Also, CNN is designed and mostly used data with 2D (Sharma, A. and Kumar D., 2020). Even though 1D data is used in the system, as seen in the results, the success of the CNN model is very close to the Deep Neural Network model and it gave very successful results.

On the other hand, the LSTM model, unfortunately, did not achieve the success of the two models here. Since LSTM is a time series system, the number of samples in the system is very important (Elsworth S., and Güttel, S.,2020). Here, it was seen that the system, which took into account the features of the previous example, could not establish a sufficient relationship with these features, and thus gave a lower result.

CHAPTER 5

CONCLUSION

RNA-Seq technology is one of the most sophisticated methods for tracking gene expressions. The gene sequences of patients can be preserved together and processed digitally thanks to these advanced technologies. The data, in particular, perform an important role in the detection of diseases such as Alzheimer's and cancers correlated to genes. A diagnosis can be produced for the next patient or a health problem can be recognized for a patient using the previous data. Because of the large dimensionality of the RNA-Seq dataset, classification and diagnosis might be quite challenging. RNA-Seq datasets from two independent data sets, Alzheimer's and Cervical Cancer, are used in this thesis. Three alternative methodologies were used to classify them.

Cervical Cancer has 741 miRNAs and 2900 samples from 50 distinct data sets, while Alzheimer's has 2.801 miRNAs and 2450 samples from 50 different data sets. The gene selection method is used, and according to it, 5%, 10%, and 30% of the total genes are selected. For this selection, the most closely related genes are chosen. Following gene selection, the system is subjected to three classifier methods: Deep Neural Network, CNN, and LSTM, with four different optimizers (Adadelta, Adam, Adamax, RMSProp), three different Learning Rates (0.001, 0.005, 0.01), three different epoch numbers (50, 75, 100), and three different batch sizes (32,64,128). For each individual, 50 data sets for Alzheimer's disease and 50 data sets for Cervical Cancer are generated. After all the train scenarios are successfully completed, average values are calculated for each disease and each gene selection case. According to this, 6 different tables show the average of each gene-selection scenario with the classifier.

A Deep Neural Network is the most accurate method for Cervical Cancer Data set training. The method produces 94.94% accuracy rate for 5% gene-selection, 95.41% accuracy rate for 10% gene-selection scenario, 94.94% accuracy rate for 30% gene-selection. 95.41% accuracies of Deep Neural Network are the most successful accuracy for this study. Correspondingly, the lowest error rate of all the studies also belongs Deep Neural Network method with 0.044. The other two error rates were 0.045 and 0.046 which are close to the lowest ones. The Deep Neural Network method is also the most robust method for two gene-selection scenarios for Alzheimer's

Dataset. The accuracies are decreased for this disease. The method outputs an 84.74% accuracy rate for 5% gene selection which is not the best accuracy for this scenario. The other two results of Deep Neural Network with an 84.79% percent accuracy rate for 10% gene selection, and 84.56% accuracy rate for 30% gene selection the most accurate method for their scenarios. In addition, the error rates of the system are 0.126, 0.110, and 0.124 for gene selection size.

After Deep Neural Network, CNN is the second most accurate approach for Cervical Cancer Dataset training. For percent 5 gene selection, the approach achieves a 92.12 percent accuracy rate, a 93.06 percent accuracy rate for percent 10 gene selection, and a 93.18 percent accuracy rate for percent 30 gene selection. The study's maximum successful accuracy for this method is 30% accuracy with a percent of 93.18. As is shown by the results, the convolutional neural network method's accuracy percentage rises as the ratio of genes selected rises. Again, After the Deep Neural Network scores for cervical, the CNN technique had the lowest error rates of all the studies. The Cervical Cancer Dataset had error rates of 0.074, 0.058, and 0.052. In the Alzheimer Dataset, the CNN approach is the most accurate for percent 5 gene-selection scenarios, with an accuracy of 84.74 percent. The approach achieves percent 83.50 and percent 83.53 accuracy for percent 10 gene selection and percent 30% gene selection, respectively, which are the second and third best scores for this scenario after the Deep Neural Network method. The Alzheimer dataset error rates are greater than the Cervical dataset error rates, as expected: 0.11, 0.11, and 0.107, respectively to gene selection size.

Finally, For Cervical Cancer datasets and Alzheimer's training, LSTM is the less accurate method. The accuracy for Cervical Cancer training is 83.41 percent with a 0.147 error score for percent 5 gene selection, the best score for this strategy. Following that, the accuracy for percent 10 gene selection is 80.59 with a 0.165 error score, and for percent 30 gene selection, the accuracy is 70.24 with a 0.264 error score. This strategy does not change the situation for the Alzheimer's dataset. In this situation, the most successful accuracy is 80.68 percent with a 0.15 error score for percent 10 gene selection. Following that, the accuracy for percent 5 gene selection is 80.05 with a 0.147 error score, and for percent 30 gene selection, the accuracy is 72.09 with a 0.189 error score.

The models generated for cervical cancer classification were applied to 50 data sets, with an average success rate of 95.41 percent in Deep Neural Network, and the best score for Alzheimer's disease was 84.79 percent with the same amount of data sets. In at least one gene selection situation, these two techniques yield at least 80% accuracy for the classifiers. Typically, Adamax and Adam optimizers are chosen with learning rates of 0.05 and 0.01 to provide the best results from these classifiers.

The fact that the model constructed using the Deep Neural Network method outperforms the others demonstrates the technology's efficacy in investigations including RNA-Seq data. Using extra datasets can help improve training and test success rates. As a result, the acquired system's reliability can be improved. The models developed as a result of the dataset access issue could be used for data on cervical cancer and Alzheimer's disease. The design of the models can be changed to explore how layers affect the categorization result. The architectures of other methods can be simplified to test whether the results depend on the architecture of the Deep Neural Network method which has the simpler one than the other two models. Lastly, these three methods can be used to classify other RNA-SEQ data sets.

RNA-Seq datasets were effectively used to create a decision support system for Alzheimer's and Cervical Cancer classification in this thesis. In conclusion, the proposed approaches for classifying cervical cancer and Alzheimer's disease in this study produced better results than in earlier studies. These approaches are proved to be useful for analyzing RNA-Seq data for specific illness types.

In addition, the sequence analysis method is widely employed in other domains. Sequence approaches are increasingly being utilized to investigate life-course and career trajectories, time consumption, organizational and national developmental patterns, speech and interaction structure, and the challenge of work/family synchronization, particularly in the social sciences and sociology. In the social sciences, this research group is described as sequence analysis. Furthermore, the usage of DNA sequencing techniques has grown in recent years, with applications in medicine, agriculture, forensic medicine, biological research, and other sectors, resulting in the growth of the pharmaceutical and chemical industries (Sripathi V. R. et al., 2021).

As a result, artificial intelligence-assisted data analysis for collected series data will play an essential role in the sector as well as the other areas indicated.

REFERENCES

Alpaydın E. (2011). Yapay Öğrenme (Artificial Learning), Boğaziçi Üniversitesi Yayınevi, Türkiye.

American Cancer Society. (2017). Cervical Cancer Fact Sheet from https://www.cancer.org/content/dam/cancer-org/cancer-control/en/booklets-flyers/cervical-cancer-fact-sheet.pd

Amin, N., McGrath A., Chen YP.P. (2019). Evaluation of deep learning in non-coding RNA classification. Nat Mach Intell 1, 246–256.

Anders S., Huber W. (2010). Differential expression analysis for sequence count data. Genome Biology, **11(10),** R106.

Asefisaray B. (2018). Uçtan uca konuşma tanıma modeli: Türkçe'deki deneyler, Doktora Tezi, Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 10-85.

Aydoğdu, S. G. M., Özsoy, Ü. (2018). Serviks Kanseri ve HPV. Androloji Bülteni, **20**, 25–29. https://doi.org/https://www.doi.org/10.24898/tandro.2018.62533

Barut A. (2008). Serviks kanserinde erken tanı ve tedavi. Sürekli Tıp Eğitimi Dergisi(Early diagnosis and treatment in cervical cancer. Journal of Continuing Medical Education).

Bateman R.J. Xiong C. Benzinger T.L. Fagan A.M., Goate A., Fox N.C. (2012). Clinical and biomarker changes in dominantly inherited Alzheimer's disease. N Engl J Med, 367-9, 795-804.

Bengio Y., Simard P. ve Frasconi P. (1994). Learning long-term dependencies with gradient descent in difficult. IEEE Transactions on Neural Networks, 5(2), 157-166.

Bezdan, T., Bačanin Džakula, N. (2019). Convolutional Neural Network Layers and Architectures. Paper presented at Sinteza 2019 - International Scientific Conference on Information Technology and Data Related Research.

Bicciato S., Pandin M., Didonè G., Di Bello C. (2003). Pattern identification and classification in gene expression data using an autoassociative neural network model. Biotechnol Bioeng. 81(5), 594-606.

Boyle P., Levin B. (2008). World Cancer Report. Ed.-International Agency for Research on Cancer (IARC), 418-424.

Braak H., Thal D.R., Ghebremedhin E., Del Tredici K. (2011). Stages of the pathologic process in Alzheimer disease: age categories from 1 to 100 years. J Neuropathol Exp Neurol, 70-11, 960-9.

Camcıoğlu Y. (2008). Adölesan çağında yeni bir aşı; human papillomavirüs aşısı. İ.Ü. Cerrahpaşa Tıp Fakültesi Sürekli Tıp Eğitimi Etkinlikleri. Adölesan Sağlığı 2 (A new vaccine in adolescence; human papillomavirus vaccine. İ.Ü. Cerrahpaşa Faculty of Medicine Continuing Medical Education Activities. Adolescent Health 2); 117- 124.

Chakraborty R., Hasija Y. (2020). Predicting MicroRNA Sequence Using CNN and LSTM Stacked in Seq2Seq Architecture. IEEE/ACM transactions on computational biology and bioinformatics, 17(6), 2183–2188.

Chen W., Lu H., Wang M. and Fang C. (2009). Gene Expression Data Classification Using Artificial Neural Network Ensembles Based on Samples Filtering, 2009 International Conference on Artificial Intelligence and Computational Intelligence, 626-628.

Cheng L., Khalitov, R., Yu, T., Yang, Z. (2022). Classificatıon of Long Sequential Data Using Circular Dilated Convolutional Neural Networks.

Chollet F. et al.,(2015). Keras. Available at: https://github.com/fchollet/keras.

Cireşan D. C., Meier U., Masci J., Gambardella L. M. (2012). Flexible, High Performance Convolutional Neural Networks for Image Classification, in Proceedings of the Twenty Second international joint conference on Artificial Intelligence, pp. 1237–1242.

Cireşan D.C., Meier U., and Schmidhuber J. (2012). Multi-column Deep Neural Networks for Image Classification.

Cloonan N., Forrest A.R.R., Kolle G., Gardiner B.B.A., Faulkner G.J., Brown M.K. (2008). Stem cell transcriptome profiling via massive-scale mRNA sequencing. Nature Methods. 7;5(7):613–619.

Cortes C., Vapnik V. (1995). Support-Vector Networks, Journal of Machine Learning, 20(3), 273-297.

Datta S., Nettleton D. (2014). Statistical Analysis of Next Generation Sequencing Data. 10.1007/978-3-319-07212-8.

Elsworth S., Güttel, S. (2020). Time series forecasting using LSTM networks: A symbolic approach. arXiv preprint arXiv:2003.05672.

Evans C., Hardin J., Stoebel D.M. (2017). Selecting between-sample RNA-Seq normalization methods from the perspective of their assumptions. Briefings in Bioinformatics.

Farfade S. S., Saberian M., Li L.-J. (2015). Multiview Face Detection Using Deep Convolutional Neural Networks, in Proceedings of the 5th ACM on International Conference on Multimedia Retrieval., pp. 643–650. Grefenstette E., Blunsom P., de Freitas N., Hermann K. M., (2014), A Deep Architecture for Semantic Parsing.

Friedman E.M., Shih R.A., Langa K.M., Hurd M.D. (2015). U.S. prevalence and predictors of informal caregiving for dementia. Health Aff, 34-10, 1637-41.

Fukushima K. (1980). Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position, Biological Cybernetics by Springer-Verlag, 36, 193-202.

Futreal, P. A., Coin, L., Marshall, M., Down, T., Hubbard, T., Wooster, R., Stratton, M. R. (2004). A census of human cancer genes. Nature Reviews Cancer, 4(3), 177–183. doi:10.1038/nrc1299

Fürnkranz J., Chan P., Craw, S., Sammut, C. et al. (2010). Mean Squared Error. 10.1007/978-0-387-30164-8_528.

Ganda R., Mahmood A. (2018). Convolutional Recurrent Deep Learning Model for Sentence Classification. IEEE Access. PP. 1-1. 10.1109/ACCESS.2018.2814818.

Gaugler J., James B., Johnson T., Marin A., and Weuve J. (2019). 2019 Alzheimer's disease facts and figures, Alzheimers Dementia, 15-3, 321-387.

Girshick R., Donahue J., Darrell T., Malik J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 580-587.

Goksuluk D. (2019). Count Based Clustering and Classification of Rna-Seq Data. PhD thesis, Hacettepe University.

Goksuluk, D., Zararsiz, G., Korkmaz, S., Eldem, V., Zararsiz, G. E., Ozcetin, E., Ozturk, A., Karaagaoglu, A. E. (2019). MLSeq: Machine learning interface for RNA-sequencing data. Computer methods and programs in biomedicine, 175, 223–231.

Gordon B.A., Blazey T.M., Su Y., Hari-Raj A., Dincer A., Flores S. (2018). Spatial patterns of neuroimaging biomarker change in individuals from families with autosomal dominant Alzheimer's disease: A longitudinal study. Lancet Neurol, 17-3, 241-50.

Goyal P. et al. (2017). Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour, arXiv preprint arXiv:1706.02677.

Guesmi, L., Fathallah, H., Menif, M. (2018). Modulation Format Recognition Using Artificial Neural Networks for the Next Generation Optical Networks. 10.5772/intechopen.70954.

Günaydın, C. (2013). Ailesinde Servikal Kanser Olan ve Olmayan Kadınların Erken Tanılama Konusundaki Davranışları. İstanbul Üniversitesi, İstanbul

Gürvit, H. İ. (2010). Demans Sendromu, Alzheimer Hastalığı ve Alzheimer Dışı Demanslar.

Hardcastle T.J., Kelly K.A. (2010). baySeq: Empirical Bayesian methods for identifying differential expression in sequence count data. BMC Bioinformatics, 11(1):422

Haykin S.S. (2009). Neural Networks and Learning Machines, 3rd ed. New York: Pearson Education, p. 906 S.

Hinton G. E. (1986). "Learning Distributed Representations of Concepts", Proceedings of the Eighth Annual Conference of the Cognitive Science Society, Amherst, Mass, Parallel Distributed Processing: Implications for

Psychology and Neurobiology, Editör: R. G. M. Morris, Oxford University Press, Oxford, UK, 46-61.

Hochreiter S., Schmidhuber J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735–80.

Hoffer E., Hubara I., and Soudry D. (2017). Train longer, generalize better: closing the generalization gap in large batch training of neural networks.

Hubel D. H. and Wiesel T. N. (1968). Receptive fields and functional architecture of monkey striate cortex J. Physiol., vol. 195, no. 1, pp. 215–243.

Hyeon J., Choi H. J., Lee K. N., Lee B. D. (2017). "Automating papanicolaou test using deep convolutional activation feature", Proc. - 18th IEEE Int. Conf. Mob. Data Manag. MDM 2017, 382–385.

ICO HPV Information Centre. (2016). Human papillomavirus and related disease, Summary Report. Available at: http://www.hpvcentre.net/statistics/reports/MYS.pdf

Ivakhnenko A. G., Lapa V. G. (1965). Cybernetic Predicting Devices, Purdue University School of Electrical Engineering.

Işık, O., Çelik, M., Keten, H. S., Dalgacı, A. F., ve Yıldırım, F. (2016). Kadın Doktorların Pap Smear Testi Konusunda Bilgi Tutum ve Davranışlarının Belirlenmesi. Cukurova Medical Journal, 41(2), 291–298. https://doi.org/10.17826/cutf.208422

Jack C.R., Lowe V.J., Weigand S.D., Wiste H.J., Senjem M.L., Knopman D.S. (2009). Serial PiB and MRI in normal, mild cognitive impairment and Alzheimer's disease: Implications for sequence of pathological events in Alzheimer's disease. Brain, 132, 1355-65.

Kalchbrenner N., Grefenstette E., Blunsom P. (2014). A Convolutional Neural Network for Modelling Sentences.

Karpathy A. (2018). Stanford University, Stanford CS class CS231n: Convolutional Neural Networks for Visual Recognition, Course Notes.

Khademi F., Jamal S. M. (2016). Predicting the 28 Days Compressive Strength of Concrete Using Artificial Neural Network. i-manager's Journal on Civil Engineering. 6.

Kim S., Kim H. (2016). A new metric of absolute percentage error for intermittent demand forecasts, International Journal of Forecasting, 32, 669-679.

Kim Y. (2014). Convolutional Neural Networks for Sentence Classification.

Kingma, D., Ba, J. (2014). Adam: A Method for Stochastic Optimization. International Conference on Learning Representations.

Kotsiantis S., Kanellopoulos D., Pintelas P. (2006). Data Preprocessing for Supervised Learning. International Journal of Computer Science. 1. 111-117.

Kukurba K.R., Montgomery S.B. (2015). RNA Sequencing and Analysis. Cold Spring Harbor protocols., 04, 2015(11):951–969.

Kumar D., Sharma D. (2019). Deep learning in gene expression modeling. In V. Balas, S. Roy, D. Sharma, P. Samui (Eds.), Handbook of Deep Learning Applications, (Smart Innovation, Systems and Technologies; Vol. 136). Springer, 136, 363.

Kurt A., Canbulat N., Savaşer S. (2013). Medical Journal of Bakırköy. Adölesan dönem cinselliğiyle öne çıkan serviks kanseri ve risk faktörleri. Bakırköy Tıp Dergisi (Medical Journal of Bakirkoy. Cervical cancer and risk factors that stand out with its adolescent sexuality. Bakirkoy Journal of Medicine), 9-2.

Kızrak M. A., Bolat B. (2018). Derin Öğrenme ile Kalabalık Analizi Üzerine Detaylı Bir Araştırma.Bilişim Teknolojileri Dergisi.

Law C.W., Chen Y., Shi W., Smyth G.K. (2014). voom: precision weights unlock linear model analysis tools for RNA-seq read counts. Genome Biology, 15(2):R29

LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553), 436–444.

Lecun Y., Bottou L., Bengio Y., Haffner P. (1998). Gradient-based learning applied to document recognition, Proc. IEEE, vol. 86, no. 11, pp. 2278–2324.

Leidinger P., Backes C., Deutscher S., Schmitt K., Mueller S.C., Frese K., et al. (2013). A blood based 12-miRNA signature of Alzheimer disease patients. Genome Biology, 14(7),R78.

Liu Y., Zhou J., White K.P. (2014). RNA-seq differential expression studies: more sequence or more replication? Bioinformatics. 02;30(3):301–304.

Love M.I., Huber W., Anders S. (2014). Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biology, 15(12):550

Maddison C. J., Huang A., Sutskever I., Silver D. (2014). Move Evaluation in Go Using Deep Convolutional Neural Networks.

Mahendran, N., Wang, Z. Hamze, F., Freitas N. (2011). Bayesian Optimization for Adaptive MCMC.

McDonald's. (2017). McDonald's Corporation Report 2017. Available at: https:// corporate.mcdonalds.com/content/dam/gwscorp/investorrelations-content/annual-reports/McDonald%27s%20 2017%20Annual%20Report.pdf.

Mishra, G. A., Pimple, S. A., Shastri, S. S. (2016). Prevention of Cervix Cancer in India. Oncology (Switzerland), 91(1), 1–7. https://doi.org/10.1159/000447575

Mortazavi A., Williams B.A., McCue K., Schaeffer L. (2008). Wold B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nature Methods, 5(7):621–628.

Newborn M. (2000). Deep Blue's Contribution to AI, Annals of Mathematics and Artificial Intelligence, 28(1–4), 27-30.

Olah C. (2015). Understanding LSTM Networks, http://colah.github.io/posts/2015-08-Understanding-LSTMs.

Oshlack A., Wakefield M.J. (2009). Transcript length bias in RNA-seq data confounds systems biology. Biology Direct., 4(1):14.

Parikh A., Miranda E.R., Katoh-Kurasawa M., Fuller D., Rot G., Zagar L. (2010). Conserved developmental transcriptomes in evolutionarily divergent species. Genome Biology, 11(3): R35.

Perez-Diez A., Morgun A., Shulzhenko N. (2007). Microarrays for Cancer Diagnosis and Classification. In: Mocellin S. (eds) Microarray Technology and Cancer Gene Profiling. Advances in Experimental Medicine and Biology, vol 593. Springer, New York, NY

Priyadarshini, I., Cotton, C. (2021). A novel LSTM-CNN-grid search-based deep neural network for sentiment analysis. The Journal of Supercomputing.

R. Collobert and Weston J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning, in Proceedings of the 25th international conference on Machine learning - ICML '08, vol. 20, no. 1, pp. 160–167.

Rabarison K.M., Bouldin E.D., Bish C.L., McGuire L.C., Taylor C.A., Greenlund K.J. (2018). The economic value of informal caregiving for persons with dementia: Results from 38 states, the District of Columbia, and Puerto Rico, 2015 and 2016 BRFSS. Am J Public Health, 108-10, 1370-7.

Rai P., Oh S., Shyamkumar, P., Ramasamy M., Harbaugh R., Varadan, V. (2013). Nano-Bio- Textile Sensors with Mobile Wireless Platform for Wearable Health Monitoring of Neurological and Cardiovascular Disorders. Journal of the Electrochemical Society. 161.

Rayavarapu K., Krishna K. K. V., (2018). Prediction of Cervical Cancer using Voting and DNN Classifiers, Proc. 2018 Int. Conf. Curr. Trends Towar. Converging Technol. ICC CT 2018, 1–5.

Reiman E.M., Quiroz Y.T., Fleisher A.S., Chen K., Velez-Pardos C., Jimenez-Del-Rio M. (2012). Brain imaging and fluid biomarker analysis in young adults at genetic risk for autosomal dominant Alzheimer's disease in the presenilin 1 E280A kindred: A case-control study. Lancet Neurol, 11-2, 1048-56.

Ritchie M.E., Phipson B, Wu D, Hu Y., Law C.W., Shi W., Smyth G. (2015). limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Research 43(7):e47 DOI 10.1093/nar/gkv007.

Robinson M.D, McCarthy DJ, Smyth G.K. (2010). edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics, 26(1):139

Robinson M.D., Oshlack A. (2010). A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biology, 11(3), R25.

Robinson M.D., Oshlack A. (2011). A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biology.,11(3): R25.

Robinson M.D., Smyth G.K. (2008). Small-sample estimation of negative binomial dispersion, with applications to SAGE data. Biostatistics, 9(2):321–332.

Rosenblatt F. (1957). The Perceptron a Perceiving and Recognizing Automaton, Cornell Aeronautical Laboratory.

Ruder S. (2016). An overview of gradient descent optimization algorithms, ArXiv e-prints, 1609.

Rukhsar, L. Bangyal, W., Khan M., Ag I., Ag A., Nisar, K., Rawat, D. (2022). Analyzing RNA-Seq Gene Expression Data Using Deep Learning Approaches for Cancer Classification. Applied Sciences. 12. 1850. 10.3390/app12041850.

Rumelhart D. E., Hinton G. E., Williams R. J. (1986). Learning Representations by Back-Propagating Errors, Nature, 323, 533-536.

S. University. (2018). CS231n: Convolutional Neural Networks for Visual Recognition, http://cs231n.github.io/neural-networks-2

Schmidhuber, J., Greff, K., Srivasava, R. K., Kutnik J. ve Steunebrink, B. R. (2017). LSTM: a search space odyssey. IEEE Transactions on Neural Networks and Learning Systems, 28(10), 2222-2232.

Selekler, K. (2003). Alzheimer ve Diğer Demanslar, Modern Tıp Seminerleri 26. Güneş Kitabevi, Ankara.

Shahbaz M., Ali S., Guergachi A., Niazi A., Umer A. (2019). Classification of Alzheimer's Disease using Machine Learning Techniques. 296-303. 10.5220/0007949902960303.

Sharma A., Kumar D. (2020). Classification with 2-D Convolutional Neural Networks for breast cancer diagnosis. arXiv preprint arXiv:2007.03218.

Shen Y., He X., Gao J., Deng L., Mesnil G. (2014). Learning semantic representations using convolutional neural networks for web search, in Proceedings of the 23rd

International Conference on World Wide Web - WWW ’14 Companion, 2014, pp. 373–374.

Spillman B., Wolff J., Freedman V.A., Kasper J.D. (2011). Informal Caregiving for Older Americans: An Analysis of the 2011 National Health and Aging Trends Study. Available at: https://aspe.hhs.gov/report/ informal-caregiving-older-americans-analysis-2011-nationalstudy-caregiving.

Sripathi V. R., Anche V. C., Gossett Z. B., Walker, L. T. (2021). Recent Applications of RNA Sequencing in Food and Agriculture.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting, Journal of Machine Learning Research, Volume 15, pp. 1929–1958.

Statnikov, A., Aliferis, C. F., Tsamardinos, I., Hardin, D., Levy, S. (2005). A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics, 21(5), 631-643.

Sultan M., Schulz M.H., Richard H., Magen A., Klingenhoff A., Scherf M. (2008). A Global View of Gene Activity and Alternative Splicing by Deep Sequencing of the Human Transcriptome. Science.,321(5891):956–960.

Sönmez Y., Nayir T., Köse S., Gökçe B., Kişioğlu A. (2012). Bir sağlık ocağı bölgesinde 20 yaş ve üzeri kadınların meme ve serviks kanseri erken tanısına ilişkin davranışları. S.D.Ü. Tıp Fak. Dergisi (Behaviors of women aged 20 and over regarding early diagnosis of breast and cervical cancer in a health center region. S.D.U. Faculty of Medicine magazine), 19(4), 124-130.

Tan, K.M., Petersen, A. Witten, D.M. (2014). Classification of RNA-seq Data.

Tieleman T., Hinton G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 4.

Trapnell C., Hendrickson D.G., Sauvageau M., Goff L., Rinn J.L., Pachter L. (2012). Differential analysis of gene regulation at transcript resolution with RNA-Seq. Nature Biotechnology. 31:46 EP.

Tuncer M. (2009). Türkiye'de kanser kontrolü. T.C. Sağlık Bakanlığı Kanserle Savaş Dairesi Başkanlığı (Cancer control in Turkey. T.R. Ministry of Health Cancer Control Department), 379- 384,

Verma, A., Ranga, V. (2018). On Evaluation of Network Intrusion Detection Systems: Statistical Analysis of CIDDS-001 Dataset Using Machine Learning Techniques, Pertanika Journal of Science and Technology, 26, 1307-1332.

Villemagne V.L., Burnham S., Bourgeat P., Brown B., Ellis K.A., Salvado O. (2013). Amyloid ß deposition, neurodegeneration, and cognitive decline in sporadic Alzheimer's disease: A prospective cohort study. Lancet Neurol, 12-4,357-67.

Wallach I., Dzamba M., Heifets A. (2015). AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery.

Walmart. (2018). Walmart: 2018 Annual Report, Available at: https://s2.q4cdn. com/056532643/files/doc_financials/2018/annual/WMT2018_Annual-Report.pdf.

Wang, W. and Lu, Y. (2018). Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model. IOP Conference Series: Materials Science and Engineering. 324. 012049. 10.1088/1757-899X/324/1/012049.

Welling K., Nanchahal K., Macdowall W., McManus S., Erens B., Mercer C., Johnson A., Copas A., Korovessis C., Fenton K., Field J. (2001). Sexual behaviour in Britain: early heterosexual experience., 358.

Witten D.M. (2011). Classification and clustering of sequencing data using a Poisson model. Annals of Applied Statistics, 5:2493–2518

Yadavendra, Chand, S. (2020). A comparative study of breast cancer tumor classification by classical machine learning methods and deep learning method. Machine Vision and Applications, 31.

Yosinski J., Clune J., Nguyen A., Fuchs T., Lipson H. (2015). Understanding Neural Networks Through Deep Visualization.

You Y., Gitman I., and Ginsburg B. (2017). Large Batch Training of Convolutional Networks.

You Y., Zhang Z., Hsieh C.J., Demmel J., Keutzer K. (2017). ImageNet Training in 24 Minutes.

Zararsiz, G., Goksuluk, D., Klaus, B., Korkmaz, S., Eldem, V., Karabulut, E., Ozturk, A. (2017). voomDDA: discovery of diagnostic biomarkers and classification of RNA-seq data. PeerJ, 5, e3890.

Zararsız G. (2015). Development and application of machine learning approaches for RNA-seq classification. PhD thesis, Hacettepe University.

Zararsız, G., Goksuluk, D., Korkmaz, S., Eldem, V., Zararsiz, G. E., Duru, I. P., Ozturk, A. (2017). A comprehensive simulation study on classification of RNA-Seq data. PloS one, 12(8), e0182507.

Zeiler M. (2012). ADADELTA: An adaptive learning rate method. 1212.