MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

PREDICTION OF CIRCRNA-DISEASE ASSOCIATIONS USING DEEP LEARNING MODELS

HACER TURGUT

MASTER THESIS

Department of Computer Engineering

Thesis Supervisor
Asst. Prof. Betül BOZ

Thesis Co-Supervisor
Asst. Prof. Beste TURANLI

ISTANBUL, 2022

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

PREDICTION OF CIRCRNA-DISEASE ASSOCIATIONS USING DEEP LEARNING MODELS

HACER TURGUT

(527619007)

MASTER THESIS
Department of Computer Engineering

Thesis Supervisor
Asst. Prof. Betül BOZ

Thesis Co-Supervisor
Asst. Prof. Beste TURANLI

ISTANBUL, 2022

ACKNOWLEDGEMENTS

I would like to express my special appreciation and deepest gratitude to my supervisor Betül Boz and my co-advisor Beste Turanlı for their continuous support and encouragement during my master thesis.

I would like to thank my parents, husband, and friends for always being there for me and supporting me.

January, 2022 **Hacer Turgut**

TABLE OF CONTENTS

ÖZET

DAİRESEL RNA-HASTALIK İLİŞKİLERİNİN DERİN ÖĞRENME METODLARIYLA TAHMİNİ

Dairesel RNA, kapalı döngü yapısına sahip tek iplikli bir RNA tipidir. Son yıllarda yapılan araştırmalar dairesel RNA'ların biyolojik süreçlerde çok önemli rolleri olduğunu ve insan hastalıkları ile ilişkili olduğunu ortaya koymuştur. Dairesel RNA'lar hastalık biyobelirteci ve ilaçlar için hedef molekül olabileceğinden dolayı, hastalıkların erken evrelerde saptanması ve hastalıkların tedavisinde kullanılması önem taşımaktadır. Bununla birlikte, dairesel RNA'lar ve hastalıklar arasındaki ilişkileri ortaya çıkarmak için yapılan geleneksel deneysel yöntemler zaman alıcı ve maliyetlidir. Bu sorunun üstesinden gelmek amacıyla, dairesel RNA'lar ile hastalıklar arasındaki ilişkilerin tahmininde kullanılabilecek ve dairesel RNA ve hastalıkların önemli özelliklerinin sayısal olarak temsil vektörlerinin oluşturulmasında hesaplamalı yöntemler geliştirilmektedir.

Çalışmalar sonucunda hesaplamalı yöntemlerin tahminleme başarısını göstermiş ve hastalıkların hangi RNA'lar ile yüksek ihtimalle anlamlı ilişkilerinin olduğunu tespit etmeyi mümkün kıldığını göstermiştir. Aynı zamanda, modellemelerde kullanılması için dairsel RNA ve hastalıkların vektörlerinin oluşturması üzerine de çalışmalar yapılmakdır. Bu çalışma, dairesel RNA ve hastalık vektörlerini oluşturmak için birden fazla veri kaynağı kullanan, derin otomatik kodlayıcı (deep autoencoder) yardımı ile dairesel RNA-hastalık çiftlerinin saklı önemli özelliklerini düşük boyutta kodlayan ve sonrasında dairesel RNA-hastalık çiftlerinin alakalılık skorunu bir derin ileri beselemeli sinir ağı ile tahmin eden DCDA ismini verdiğimiz bir derin öğrenme tabanlı dairesel RNA-hastalık ilişkisi tahmin metodolojisi önermektedir. Bu alandaki çalışmalarda kullanılan veri setinde yaptığımız 5 katmanlı çapraz doğrulama sonuçları, modelimizin 0.9794 AUC skoru ile literatürdeki en gelişmiş tahmin yöntemlerinden daha iyi performans gösterdiğini göstermiştir.

Ocak, 2022 **Hacer Turgut**

ABSTRACT

PREDICTION OF CIRCRNA-DISEASE ASSOCIATIONS USING DEEP LEARNING MODELS

Circular RNA is a single-stranded RNA with a closed-loop structure. In recent years, researches revealed that circular RNAs have crucial roles in biological processes and are related to human diseases. Finding out potential circRNAs as disease biomarkers and drug targets is crucial since it can lead to detecting diseases in the early stages and be used to treat humans. However, in conventional experimental methods, doing experiments to reveal associations between circular RNAs and diseases is time-consuming and costly. In order to overcome this problem, various computational methodologies are proposed to extract essential features for both circular RNAs and diseases and predict the associations. Studies showed that computational methods successfully predicted performance and made it possible to detect possible highly related circular RNAs for diseases. This study proposes a deep learning-based circRNA-disease association predictor methodology called DCDA, which uses multiple data sources to create circRNA and disease features and reveal hidden feature codings of a circular RNA-disease pair with a deep autoencoder, then predict the relation score of the pair by a deep neural network. 5-fold cross-validation results on the benchmark dataset showed that our model outperforms state-of-the-art prediction methods in the literature with the AUC score of 0.9794.

January, 2022 **Hacer Turgut**

SYMBOLS

$\boldsymbol{\lambda}$	: Regularization parameter
$\mathbf{R}$	: Set of data
$\boldsymbol{\Delta}$	: Dissimilarity function
$\lvert \boldsymbol{a} \rvert$	: Lenght of sequence a
$\boldsymbol{tail(a)}$	: Sequence a without the first character
$\boldsymbol{TP}$	: True positive
$\boldsymbol{TN}$	: True negative
$\boldsymbol{FP}$	: False positive
$\boldsymbol{FN}$	: False negative
$\mathbf{exp}(\boldsymbol{x})$	: Exponential of x
$\boldsymbol{std}$	: Standard deviation
$\boldsymbol{c_i}$	: ith circRNA in data
$\boldsymbol{d_i}$	: ith disease in data
$\boldsymbol{lev(x, y)}$	: Levenstein distance between x and y sequences
$\boldsymbol{seq_{c_i}}$	: Sequence of ith circRNA
$\boldsymbol{CD}$	: CircRNA-disease association matrix
$\boldsymbol{DC}$	: Disease-circRNA association matrix (transpose of CD)
GIP_CD	: CircRNA-circRNA GIP kernel similarity matrix
GIP_CM	: CircRNA-miRNA GIP kernel similarity matrix
SIM_CC	: CircRNA-circRNA sequence similarity matrix
GIP_DC	: Disease-disease GIP kernel similarity matrix
GIP_DM	: Disease-miRNA GIP kernel similarity matrix
SIM_DD	: Disease-disease semantic similarity matrix

ABBREVIATIONS

A	: Adenine
AUC	: Area Under the ROC Curve
C	: Cytosine
circRNA	: Circular RNA
CNN	: Convolutional Neural Network
CV	: Cross-Validation
DAE	: Deep Autoencoder
DAG	: Directed Acyclic Graph
DNN	: Deep Neural Network
DO	: Disease Ontology
ELM	: Extreme Learning Machine
G	: Guanine
GAN	: Generative Adversarial Network
GIP	: Gaussian Interaction Profile
GO	: Gene Ontology
HMDD	: Human MicroRNA Disease Database
LLC	: Locality-constrained Linear Coding
LMT	: Logical Model Tree
lncRNA	: Long non-coding RNA
MAE	: Mean Absolute Error
miRNA	: Micro RNA
MSE	: Mean Squared Error
NLM	: National Library of Medicine
PCA	: Principal Component Analysis
piRNA	: Piwi-interacting RNA
RBP	: RNA Binding Protein
ReLU	: Rectified Linear Unit
RF	: Random Forests
RNA	: Ribonucleic Acid
ROC	: Receiver Operating Characteristic

SGD : Stochastic Gradient Descent

siRNA : Small interfering RNA

SVM : Support Vector Machine

U : Uracil

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Circular RNA (circRNA) is a single-stranded RNA with a continuous loop structure resulting from its covalently joined 5' and 3' ends. Due to its closed-loop structure, circRNAs are generally stable [1]. Even though researchers discovered circRNAs in the 1970s [2, 3], their significance was unrecognized and considered as RNA splicing error. Thanks to improvements in RNA sequencing technology, bioinformatics, and many studies, many circRNAs are discovered. Researchers established various functionalities of circRNAs in biological processes such as acting as sponges for microRNAs, regulating gene transcription, and expression by interacting RNA Binding Proteins (RBPs) [4]. In recent years, circRNAs are seen as potential biomarkers of several diseases and treatment targets. For example, [5] found out the correlation between tumor size and the level of expression of hsa_circ_0012673, hsa_circ_0000064, circPUM1, hsa_circ_0007385, and circ_000073558 in lung carcinoma tissue. A number of researchers collected circRNA-disease associations from previous research papers and created circRNA related databases, like circR2Disease [6], Circad [7], circ2Disease [8], and CircFunBase [9]. As the number of expressed associations between diseases and circRNAs increases, the opportunity of predicting novel circRNA-disease relationships by computational algorithms becomes possible. Recently, various machine learning and deep learning-based methods are proposed. In the KATZHCDA model, Fan et al. [10] used similarity scores between circRNA expression profiles for circRNA representations and disease phenotype similarity for disease representations. From known circRNA-disease associations, the Gaussian Interaction Profile (GIP) kernel is applied, and features are generated for both circRNAs and diseases. A graph-based method, KATZ, is used as the prediction algorithm. Wang et al. [11] also used the GIP kernel to extract features for disease and circRNAs and the semantic similarity between diseases. They used Convolutional Neural Network (CNN) as a feature extraction method and Extreme Learning Machine (ELM) as the predictor. In MSFCNN model [12], four circRNA and seven disease similarity feature based on topological and biological data from various sources and circRNA-disease associations predicted by two-dimensional CNN. In [13], the proposed method used the same features in [11]. Generative Adversarial Network (GAN) is used as a feature extraction method and Logical Model Tree (LMT) classifier. Zheng et al. [14]

used gene associations of circRNAs and circRNA sequence similarity using Chaos Game Representation and used Support Vector Machine (SVM) as the prediction model. Ge et al. [15] proposed a circRNA representation method that looks circRNAs' gene targets and creates a circRNA vector based on semantic similarity between other circRNAs' gene targets. Disease vectors are created based on disease semantic similarity between diseases. circRNA and disease vectors are reconstructed using Locality-constrained Linear Coding (LLC) to encode vectors such that vectors can preserve local information in previous information. In addition to these features, unlike other researches, in this study, the cosine similarity function is applied to get circRNA and disease similarity scores from circRNA-disease relationships instead of GIP. Label propagation methodology is appliedto all similarity networks. The average score between circRNAs and diseases is acceptedas the final prediction score. In AE-DNN model [16], circRNA vectors are generated from sequence similarity by edit distance, and disease vectors are generated from semantic similarity using disease ontology relationships. GIP kernel is also used to get additional features for both circRNAs and diseases. circRNA and disease features are concatenated by taking the average. To extract essential features from all features, the autoencoder model is used. A deep neural network model is used to predict association scores between circRNAs and diseases. Deepthi and Jereesh introduced a methodology called AE-RF [17] that feeds information gathered from the circRNA-disease association, circRNA functional similarities, and disease semantic similarities to an autoencoder model to extract hidden biological patterns and predict circRNA-disease associations with Random Forests (RF) classifier. Wei et al. proposed a model called iCircDA-MF [18]. They constructed features using disease semantic information, circRNA-gene, gene-disease, and circRNA-disease data. In order to detect and correct false-negative associations, neighbor interaction profiles based on the circRNA similarity and disease similarity used and circRNA-disease associations are predicted by matrix factorization. Want et al. [19] formed a unified descriptor of circRNA-disease pairs by disease semantic similarity information and GIP kernel similarity information of diseases and circRNAs. They proposed to use fast learning with graph convolutional networks to reveal high-level features from unified descriptors then predicted new circRNA-disease pairs with Forest by Penalizing Attributes classifier.

In this study, we propose a deep learning-based methodology called DCDA that consists of two parts: a circRNA-disease association feature extraction model and a circRNA-disease association prediction model. This methodology uses multiple data to generate circRNA-disease association features, these are manually curated circRNA-disease associations, disease-disease semantic similarity information, and disease-miRNA associations. In order to have a consistent overall dataset, we generated synonym dictionaries for both diseases and circRNAs. The experimental results revealed that DCDA has the highest AUC score on the CircR2Disease dataset comparing other state-of-the-art methodologies.

1.1. Contributions of the Thesis

In recent years, circRNAs are seen as potential biomarkers of several diseases and treatment targets. Using circRNAs as biomarkers or therapeutic targets needs to be further investigated due to their complex roles. Based on these characteristics, circRNAs are likely to guide the development of new diagnostic and therapeutic strategies as well as the prevention of diseases. Moreover, finding out potential circRNAs as disease biomarkers and drug targets is very crucial since it would decrease the potential RNA molecules that should be examined. Several researchers proposed various machine learning, and deep learning-based models that are used for both feature extraction and circRNA-disease association prediction. Those studies showed that deep learning methods increase the prediction performance of circRNA-disease interactions. In this study, our aims are:

- building a model that efficiently predicts which diseases are associated with which circRNAs,
- providing direction for the development of new and accurate diagnostic and/or prognostic biomarkers and has the potential to aid in the design of personalized medical approaches to complex human diseases in the future,
- after applications of our proposed study, other new classes of non-coding RNAs such as piwi-interacting RNAs (piRNAs), Small interfering RNA (siRNAs), etc. may be used to find their potential associations in diseases. We will create an exemplary workflow for other types of RNAs as well.

To reach our aims, we will use different methodologies. These are:

- using a deep autoencoder to reveal hidden feature combinations,

- extracting representative circRNA and disease feature vectors in low dimensional space,
- exploring a deep learning model that has high prediction performance for circRNA-disease interactions,
- using miRNA-disease information as a validation of our proposed interactions,
- validating our model's performance on different circRNA-disease datasets.

1.2. Outline of the Thesis

The rest of the thesis is organized as follows. In Chapter 2, circular RNAs, associations between circular RNA and diseases, previous methodologies used in the prediction of association between circular RNAs and diseases, circular RNA-disease association databases, and auxiliary databases used in the area are explained. In chapter 3, background information about the proposed methodology is explained such as feature generation methods, deep learning model structure and model parameters, useful methodologies in neural network models, and performance metrics. In chapter 4, our proposed model will be described including data preparation, model structure, model parameters, and used methodologies. In chapter 5, experimental results during model parameter optimization and feature selection, the performance of the model on two commonly used datasets, comparison of the methodology with state-of-the-art methods, and examination on predicted circRNA-disease pairs are performed. Finally, the last chapter is about methodology overview, discussing results and the future work.

2. RELATED WORK

2.1. Circular RNA-Disease Associations

Ribonucleic acids (RNAs) are complex polymeric compounds that play essential biological activities such as encoding and decoding genetic information, regulatingthe environment during gene transcription, catalyzing chemical reactions, and making proteins by joining amino acids. There is a type of functional RNA group callednon-coding RNAs that do not have the capability of encoding proteins but has essential roles during gene expressions. Twenty years ago, a type of non-coding RNA, microRNAs, was discovered, and in recent years long non-coding RNAs (lncRNAs) are investigated [20]. Even though researchers discovered circRNAs in the 1970s [2, 3], their significancewas unrecognized and considered as RNA splicing error. The organism in which researchers first noticed circRNA are pathogens [2]. In the early 2000s, researchers discovered that circRNAs are also produced in rat, and human organisms [21, 22, 23]. Thanks to improvements in sequencing technology and change in biological approaches after the discovery of the importance of non-coding RNAs, circRNAs are put under the scope and thousands of circRNAs are discovered [24, 25, 26]. Nowadays, thousands of circRNAs are discovered in diverse eukaryotic species.

Unlike linear RNA, circular RNA (circRNA) is a single-stranded RNA with a continuous loop structure resulting from its covalently joined 5' and 3' ends. Due to its closed-loop structure, circRNAs are generally stable [1]. circRNAs comprise from different ways such as from introns, from exons, from both from exon and introns, and tRNAs. ecircRNA, ciRNA and EIciRNA, tricRNA are examples of these formation types, respectively [27, 28, 29, 30]. Most of the circRNAs are formed as a result of back-splicing and comprise from exons [24]. Figure 2.1 shows different types of circRNA formations.

Numerous circRNAs have been found to be associated with diseases. Wang et al. [5] reported the correlation between tumor size and the level of expression of hsa_circ_0012673, hsa_circ_0000064, circPUM1, hsa_circ_0007385, and circ_000073558 in lung carcinoma tissue. Yang et al. [31], compared glioblastoma clinical samples with other tumor-adjacent tissues and found out that Circ-FBXW7 is reduced in glioblastoma samples. Liang et al. [32] discovered circRNA circ-ABCB10 in

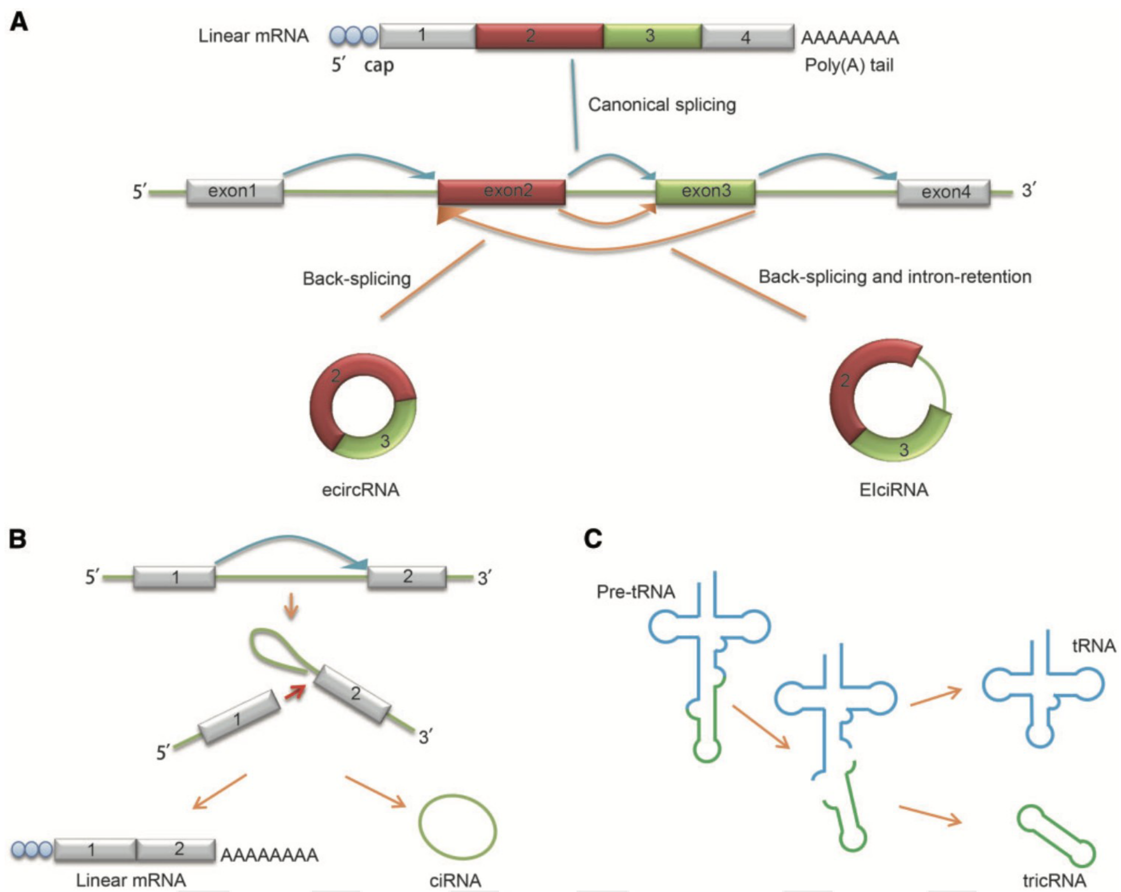


Figure 2.1: Different type of circRNA formations. (A) Exonic circRNA and exon–intron circRNA. (B) Circular intronic RNA. (C) tRNA intronic circRNA [24].

breast cancer cells and how it promotes breast cancer by sponging miR-1271.

As the importance of circRNAs come to light, the computational power of computers developed and computational models improved circRNA-disease association prediction algorithms became prevalent. As some of the methodologies are based on classical statistical models, some of them are based on graph and machine learning-based. In recent years, researchers in the area applied deep learning algorithms to explore novel circRNA-disease associations.

2.2. Circular RNA-Disease Association Databases

A number of researchers collected circRNA-disease associations from previous researches and created circRNA related databases. These databases are public and available online. Some of these databases are circR2Disease [6], Circad [7], circ2Disease

[8], and CircFunBase [9].

2.2.1. CircR2Disease

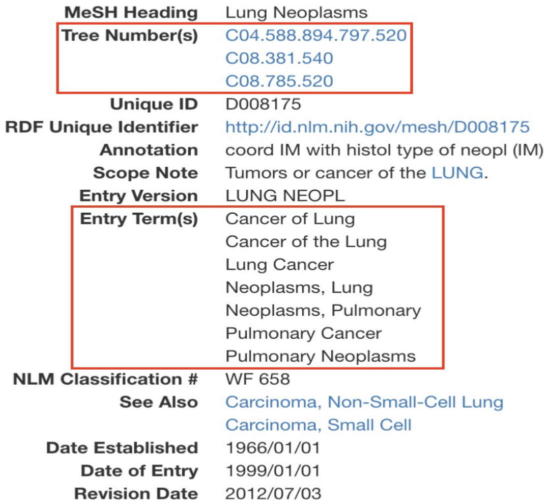
CircR2Disease database [6] contains experimentally supported associations between circRNAs and diseases. The current version of the database has 739 entries which include 661 unique circRNAs and 100 diseases. The database also includes circRNA name, coordinates and gene symbol of circRNAs, disease names, expression patterns of circRNAs (upregulated or downregulated), information of how a circRNA is related to a disease, the PubMed ID, and the year of the publication, whichis the evidence of the association. CircR2Disease database is available through http://bioinfo.snnu.edu.cn/CircR2Disease/ the link. Database website has an easy-to-use user interface with browse, search and download functionalities.

2.2.2. Circ2Disease

Circ2Disease database [8] is published in 2018. Circ2Disease database has manually curated associations between circRNAs and human diseases collected from reliable 120 publications. There are 273 associations between 273 circRNAsand 54 human diseases. Each association is shown with circRNA name, disease name, expression pattern, method of the experimentation, description of the association, and some extra information. The database is available through http://bioinformatics.zju.edu.cn/Circ2Disease/index.html.

2.2.3. CirFunBase

CirFunBase [9] is a web-based database published in 2019. CirFunBase is the first database that has circRNA-disease relationships in species other than humans such as Mus musculus. The database contains both experimentally validated and predicted circRNA functionalities. In the last version, there are more than 7000 manually curated circRNA entries. CirFunBase also has circRNA-miRNA interaction network visualizations. The database website is http://bis.zju.edu.cn/CircFunBase/.



MeSH Heading	Lung Neoplasms
Tree Number(s)	C04.588.894.797.520 C08.381.540 C08.785.520
Unique ID	D008175
RDF Unique Identifier	http://id.nlm.nih.gov/mesh/D008175
Annotation	coord IM with histol type of neopl (IM)
Scope Note	Tumors or cancer of the LUNG.
Entry Version	LUNG NEOPL
Entry Term(s)	Cancer of Lung Cancer of the Lung Lung Cancer Neoplasms, Lung Neoplasms, Pulmonary Pulmonary Cancer Pulmonary Neoplasms
NLM Classification #	WF 658
See Also	Carcinoma, Non-Small-Cell Lung Carcinoma, Small Cell
Date Established	1966/01/01
Date of Entry	1999/01/01
Revision Date	2012/07/03

Figure 2.2: Example search result of "lung cancer" term in MeSH database. Tree numbers and entry terms are marked with a red box.

2.2.4. Circad

Circad database [7] is published in 2020. The database contains manually curated circRNA-disease associations. There are 1388 circRNAs, 150 diseases, and 720 genes on their website http://clingen.igib.res.in/circad/.

2.3. Auxiliary Databases

Researchers use auxiliary databases to extract useful information about circRNA and diseases. MeSH and HMDD datasets are very popular in studies.

2.3.1. circBank

CircBank [33] is an extensive database of human circRNA. There are more than 140,000 human-annotated circRNA that are gathered from multiple sources. The database contains basic information of circRNA such as chromosome location and host gene as well as predicted related miRNAs. The database is publicly available at circbank.cn.

2.3.2. circBase

CircBase [34] is a database that provides users a single unified circRNA information database with a simple web-based interface. It is possible to search circRNAs by names or their sequence and download datasets in a variety of file types. We used this database to collect circRNA sequences. The database is available at http://www.circbase.org/.

2.3.3. MeSH

MeSH database [35] is provided by the National Library of Medicine (NLM). MeSH is a vocabulary for indexing medical and biological publications. The database also has disease-specific information and Directed Acyclic Graphs (DAGs) generated from relationships between diseases. Using these DAGs, tree numbers are assigned to diseases. For example, Figure 2.2 illustrates some information from lung neoplasms disease's page. Tree Numbers(s) are numbered format of lung neoplasms' DAGs. If we split a tree number by dot symbols, we will end up with nodes of the DAG where the first node is the top level of the DAG and the last node is the disease node. Diseases can be searched through https://meshb.nlm.nih.gov/search page.

2.3.4. HMDD

The Human microRNA Disease Database (HMDD) [36] is a human miRNA-disease relationship database that is derived from experimentally validated researches. HMDD is first published in 2017 and the last version (HMDD v3.2) is released in 2019. In the last version, there are 35547 miRNA-disease association entries, 1206 unique miRNA genes, and unique 893 diseases. In this study, we used HMDD v3.2.

3. BACKGROUND INFORMATION

3.1. Interaction Profile

An interaction profile is a binary vector that has interaction and the absence of an interaction between an element with other elements in a network. For example, in a disease-circRNA association network, if there is an association between a disease and a circRNA, the corresponding instance in the vector is 1, otherwise 0. Figure 3.1 shows how disease interaction profiles and circRNA interaction profiles are generated.

3.2. Gaussian Interaction Profile Kernel

GIP kernel is a way of constructing an interaction kernel to find similarities in a network. GIP kernel, also known as Radial Basis Function (RBF) kernel. Laarhoven et al. [37] proposed to build a kernel from interaction profiles of drug-target protein associations mentioned in Section 3.1. The idea behind this is that drugs associated with similar target proteins will also interact with similar target proteins.

GIP kernel similarity between element *x* and element *y* is calculated as follows:

$$\mathrm{GIP}(\mathrm{x}, \mathrm{y}) = exp(-\lambda \parallel I(x) - I(y) \parallel^2) \tag{3.1}$$

$$\lambda = \frac{1}{\frac{1}{n}\sum_{i=0}^{n} \parallel I(i) \parallel^2} \tag{3.2}$$

Where $I(x)$ is the interaction profile matrix, and $I(x)$ and $I(y)$ represent interaction profiles of element x and element y respectively. λ is the regularization parameter that is used to control the kernel bandwidth. n stands for the total number of unique elements.

3.3. Sequence Similarity Method: Levenstein Distance

CircRNAs are composed of small elements called ribonucleotide bases: adenine (A), cytosine (C), guanine (G), and uracil (U). A circRNA can be represented via its sequence of elements. Similar patterns between two circRNA sequences can reveal similar functionalities between circRNAs. Looking at sequence information of circRNAs to reveal similarities is a common technique. Several sequence similarity functions are used

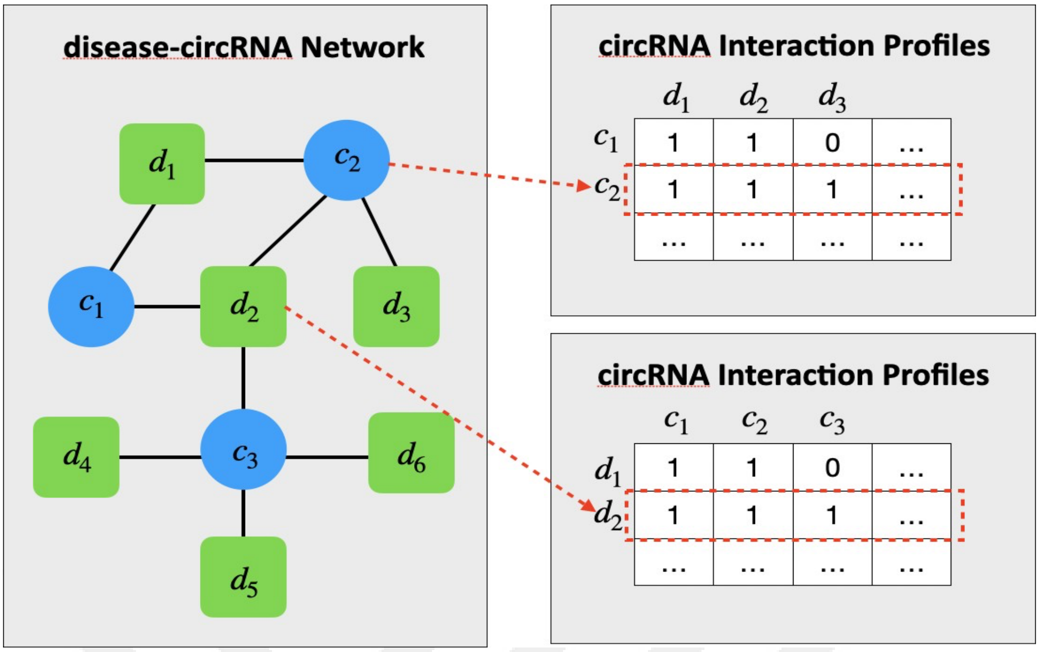


Figure 3.1: Construction of disease interaction profiles and circRNA interaction profiles from a disease-circRNA network. Squares are diseases and circles are circRNAs.

in the literature such as Cosine, Jaccard, Levenstein, and Euclidean distance algorithms. In this section, we are going to present the Levenstein distance sequence similarity algorithm.

Levenstein distance algorithm measures the difference between two sequences by looking at the minimum number of character edits to transform one sequence to the other sequence. Character-based edits are insertions, deletions, and substitutions. An example of edit operations between two sequences is shown in Figure 3.2.

The Levenstein distance between two sequences a and b is described in Equation 3.3.

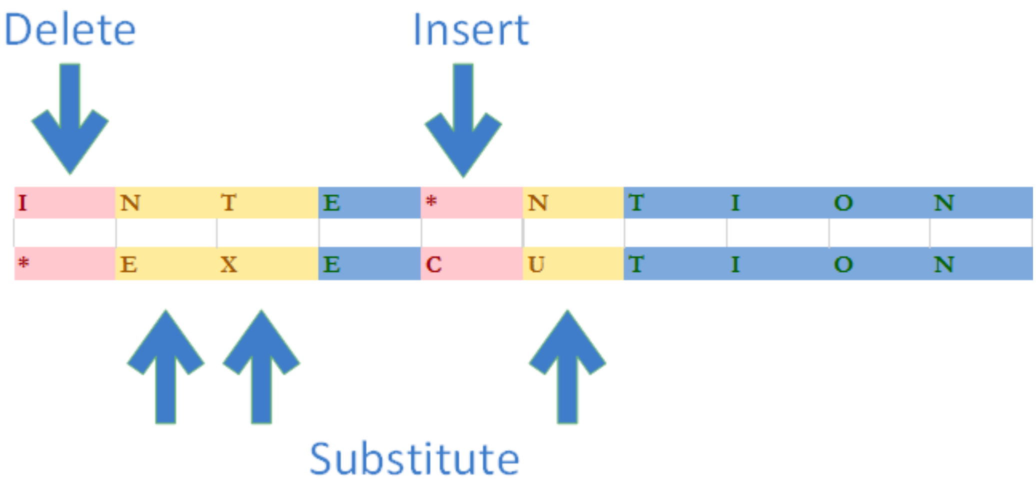


Figure 3.2: An example of edits to make one sequence similar to the other sequence in Levenstein distance algorithm [38].

$$lev(a,b) = \begin{cases} |a|, & if\ |b| = 0, \\ |b|, & if\ |a| = 0, \\ lev(tail(a), tail(b)), & if\ a[0] == b[0], \\ 1 + \min \begin{cases} lev(tail(a), b) \\ lev(a, tail(b)) \\ lev(tail(a), tail(b)) \end{cases} & otherwise \end{cases} \tag{3.3}$$

3.4. Disease Semantic Similarity

Semantic similarity is a method of calculating the similarity of documents or words based on their content/meaning. Wang et. al. proposed a method to calculate semantic similarity between genes by their Gene Ontology (GO) terms [39]. GO terms are generated based on directed acyclic graphs (DAGs) where nodes are genes and edgesare relationships between genes. Based on the same idea, Li et al. used Disease Ontology(DO) terms to calculate semantic similarity between diseases where DO terms [40].

The semantic similarity between disease d_i and disease d_j is calculated as follows:

$$\text{SemanticSimilarity}(d_i, d_j) = \frac{\sum_{e \in N_{d_i} \cap N_{d_j}} (D_{d_i}(e) + D_{d_j}(e))}{DV(d_i) + DV(d_j)} \tag{3.4}$$

$$DV(d) = \sum_{e \in N_d} D_d(e) \quad (3.5)$$

$$D_d(e) = -\log\left(\frac{num(DAGs(e))}{num(diseases)}\right) \quad (3.6)$$

Where $num(DAGs(e))$ represents the number of DAGs containing disease e, $num(diseases)$ represents a number of all diseases. If there are no common trees between diseases, their similarity score is set to 0.

3.5. Deep Learning

3.5.1. Neural Networks

Neural networks are part of machine learning algorithms. The neural network structure is inspired by the human brain. Neural networks provide machine learning with interconnected neurons, just as the learning activity takes place with the signals between the neuron networks in the human brain.

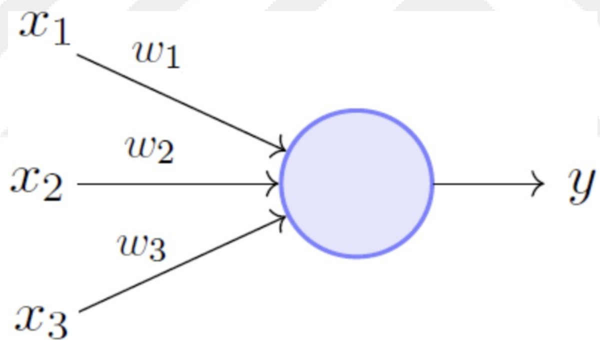


Figure 3.3: Perceptron model [41].

The most basic type of neural network, perceptron, is proposed in 1969 [42]. The perceptron model is a single-layer model that has an input and output. As shown in Figure 3.3 in inputs are *x* values, *w* values are weights and y is the output value. Model learning is based on the following equation and the learned function is represented as *f*:

$$f(x) = w.x + b \quad (3.7)$$

Where *b* is the bias value. The model aims to find optimal weights and bias values such that predicted values are close enough to the real y values.

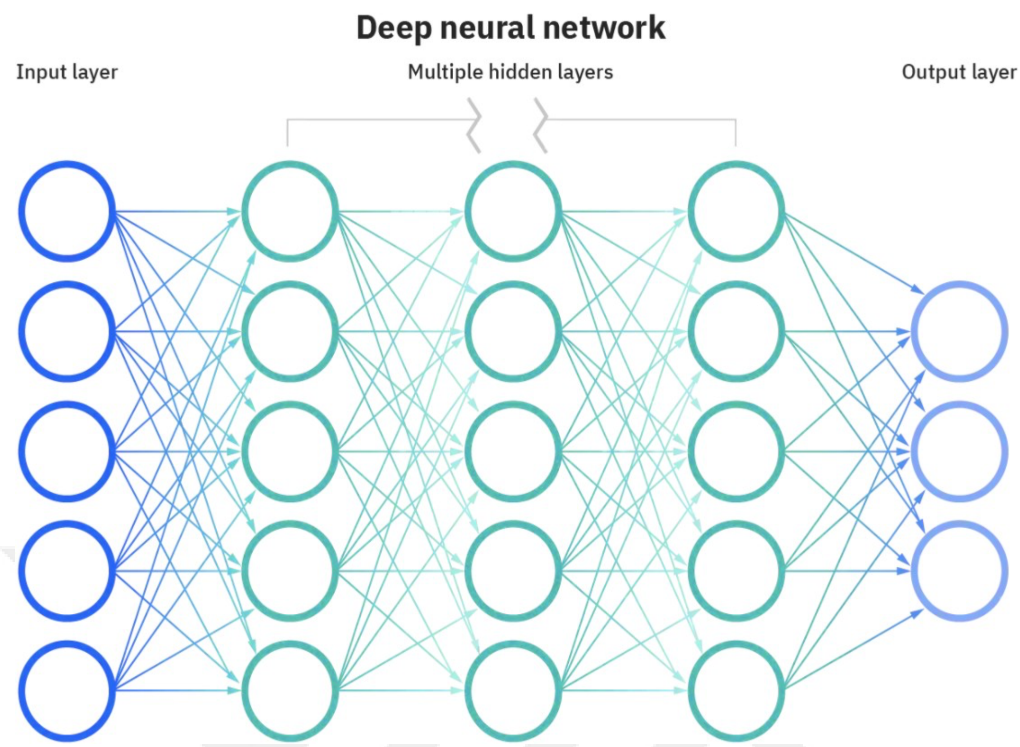


Figure 3.4: Deep neural network [43].

There are a considerable amount of neural network architectures. Some of the architectures are feed-forward networks that mean the model does the learning operation straight ahead from the input to output layer without having a backward connection. Recurrent neural networks (RNNs) are different from feed-forward neural networks, they can have a connection with past nodes and can have a looped structure.

Deep neural networks (DNNs) are more complex structures than perceptrons. As shown in Figure 3.4, DNNs have multiple layers and can have multiple neurons in layers. We can consider each node in a DNN as a regression function. Connections between nodes have weight and these weights show the importance of the node. Inputs of a node are multiplied with the corresponding weight and then summed. The output of this operation is fed into an activation function which determines the final output of the node. There are several activation functions such as sigmoid, tanh, and ReLU. Figure 3.5 shows some of activation functions' formulations and their line graphs.

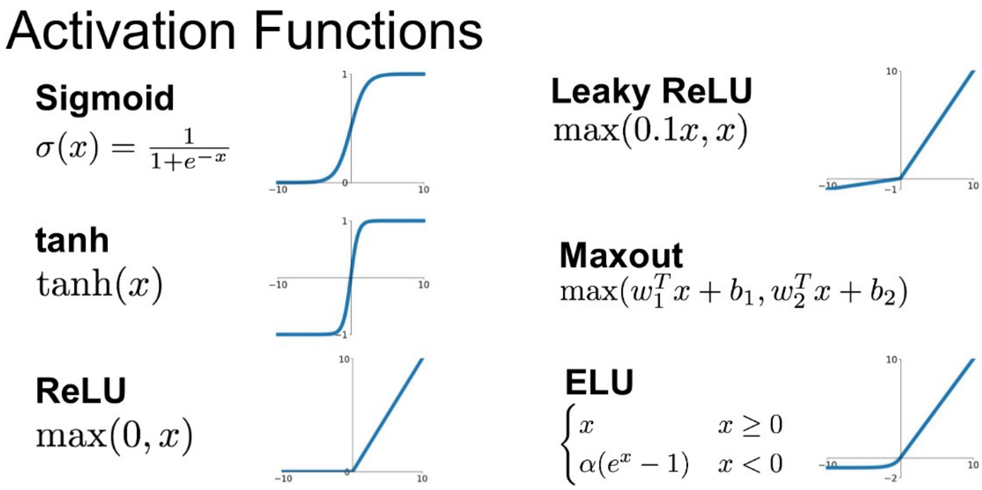


Figure 3.5: Different activation functions and their graphs [44].

Some activation functions enable neural networks to converge faster. For example, while the sigmoid function has exponential operation ReLU function is not computationally expensive and converges quickly. ReLU is the most preferred activation function because of its speed and simplicity while running backpropagation. The sigmoid function is generally preferred for classification problems since its output range is between 0 and 1.

The most important thing in a neural network is learning optimal weights and biases. In the first phase, weight and biases are initialized with random values. Optimization of weight and biases is accomplished by backpropagation. Firstly, the estimation is made with the current weight and biases, and the difference between the real value and the prediction is calculated with the cost function. Which way to update weights and biases is answered by finding the gradient by taking the derivative of the cost function. Weights and biases are updated according to the opposite direction of the gradient such that it minimizes the loss calculated by the cost function.

There are several types of cost functions used based on the problem. If the problem is regression then the cost function should be appropriate for real values for example mean squared error (MSE), and mean absolute error (MAE). For classification problems, there is a cost function that calculates the probability of an instance being part of a class. The most common cost function for classification problems is cross-entropy. When cross-entropy is used for binary classification problems it is called binary cross-entropy also known as log loss. Binary cross-entropy is calculated as follows:

$$-\frac{1}{N}\sum_{i=0}^{N} y_i \cdot \log\big(p(y_i)\big) + (1-y_i)\cdot \log\,(1-p(y_i)) \tag{3.8}$$

Where N is the number of instances, y_i is the real label, and $p(y_i)$ is the predicted label of the i^{th} sample.

Optimizers are functions that adjust how the model should learn with backpropagation. Stochastic Gradient Descent (SGD) is one of the most well-known optimization functions. In this method, a mini-batch, in other words, a small sample of the input, is taken as the input. Weights are initialized with a small number such as 0.1 and biases with zeros. After a one-time feed-forward iteration, the loss is calculated with a cost function. The gradient is calculated by taking the derivative of the cost function. Weights and biases are updated by the opposite direction of the gradient since we want to minimize the loss. Adam optimizer [45], which is an extensive version of SGD, is oneof the state-of-the-art optimizers used widely in machine learning applications. Adam optimizer is preferred because of its efficiency in computation, low memory usage, and is appropriate for non-stationary and sparse samples. The algorithm uses an adaptive learning rate instead of a static learning rate and accelerates finding the local minimum during gradient descent. Since The algorithm use exponentially weighted average and exponential moving average of the gradients, it solves sparse gradient problem and works well on non-stationary data.

3.5.2. Autoencoders

Autoencoders are first proposed in Rumelhart et al.'s [46] study. An autoencoder is a type of neural network that composes an encoder and decoder parts such that the middle layer in the network learns an informative representation of the input data in a supervised way. The encoder part compresses the input data into a small dimensional vector, and the decoder part transforms the compressed data to the original input data as far as possible. As described in [47], the aim of the autoencoder is learning $A : \mathbb{R}^n \longrightarrow \mathbb{R}^p$ (encoder part) and $B : \mathbb{R}^p \longrightarrow \mathbb{R}^n$ (decoder part) such that

$$\arg\min_{A,B} = \mathrm{E}[\Delta(\mathrm{x}, \mathrm{B} \circ \mathrm{A}(\mathrm{x}))] \tag{3.9}$$

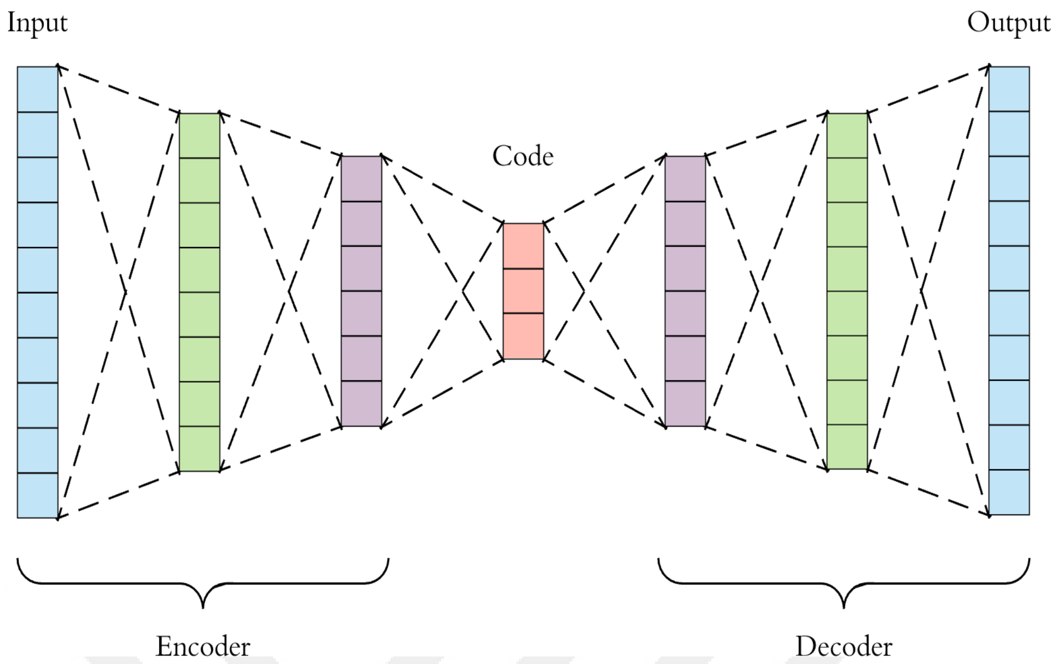


Figure 3.6: Structure of a sample deep autoencoder [48].

$\mathbb{R}^n$ is the input set, $\mathbb{R}^p$ is the encoded set, A is the class of functions from $\mathbb{R}^n$ to $\mathbb{R}^p$ and B is the class of functions from $\mathbb{R}^p$ to $\mathbb{R}^n$. n and p are positive integers. E is the expectation over the distribution of x, Δ is the dissimilarity function which is used to calculate the similarity between the input and the output of the decoder.

A simple autoencoder composes of an input layer, a hidden layer, and an output layer. In our study, we are using a deep autoencoder similar to Figure 3.6. Adding a hidden layer in the encoder and decoder parts makes a model more complex and enablesthe model to discover complex relationships between features.

3.6. Overfitting and Underfitting Problems and Their Solutions

Statistical and machine learning models can have various problems such as overfitting and underfitting. Overfitting occurs when the model learns the training data so well but can not have a good performance on the test data because of its poor generalization. Underfitting occurs when the model can not capture the trend and characteristics of the data so that does not fit the data well. Since overfitting and underfitting problems can occur, researchers split their data into train and test sets so that they can check the model performance. To make testing more reliable cross-validation (CV) methodology is mostly used. There are three variations of CV, these are k-fold,

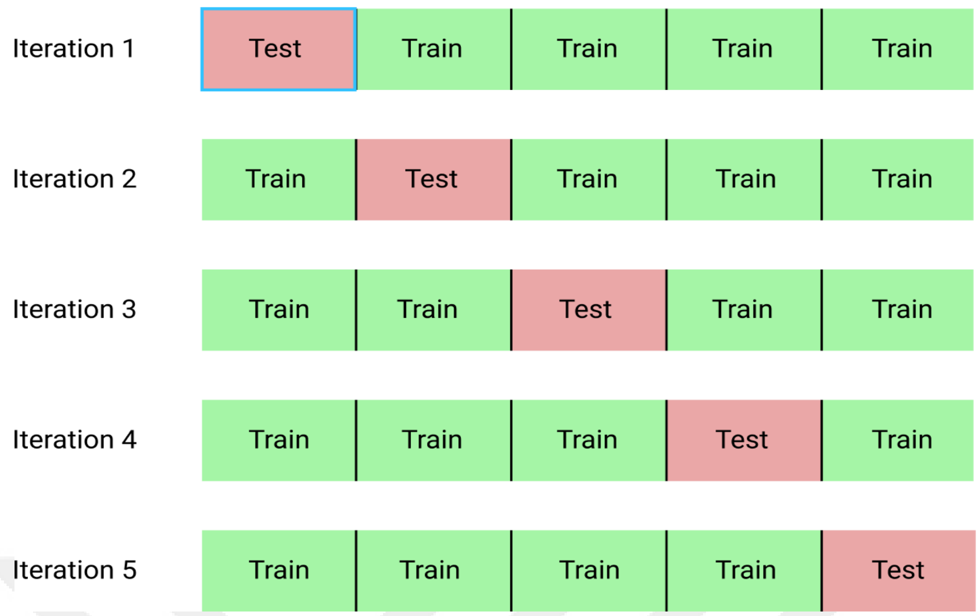


Figure 3.7: 5-fold cross-validation train-test split example [49].

stratified, and leave-one-out CV methods. In the K-fold CV method, data is split into K parts where K-1 of the parts is used in training and the remaining 1 part is used in testing. This process is repeated until each part from K parts is used as the test set. Figure 3.7 shows an example of 5-fold CV's train test splits.

Stratified CV has the same logic as k-fold CV but it splits data using a stratified random sampling method. The stratified random sampling method [50] generates subsets of the data with the same proportions of labels in subsets. For example, for a binary classification problem, data will be split into K parts where the proportion of categories are the same in each part.

To resolve the overfitting problem, early-stopping is another widely used method. The early-stopping function stops the model training after some iterations where the loss of the test data starts increasing. In this method, the maximum number of iterations and the maximum number of consecutive increasing loss results are going to be toleranced should be set.

3.7. Performance Metrics

The area under the receiver operating characteristics curve (AUC) [51] is used to compare overall performance. Receiver operating characteristic (ROC) [52] curves figures display the performance based on true positive and false-positive rates of the model. Accuracy, precision, recall, and F-1 metric formulas are described below.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{3.10}$$

$$Precision = \frac{TP}{TP+FP} \tag{3.11}$$

$$Recall = \frac{TP}{TP+FN} \tag{3.12}$$

$$F-1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.13}$$

4. DCDA: DEEP LEARNING BASED CIRCULAR RNA-DISEASE ASSOCIATION PREDICTION METHODOLOGY

In this study, we propose a methodology called DCDA to predict the association between circRNAs and diseases based on multiple data sources and deep learningmodels. Multiple datasets are gathered from different sources including circRNA-disease association databases, circRNA related datasets, and disease-related datasets. After data collection, our methodology has five main steps as illustrated in Figure 4.1. Firstly, circRNA and disease synonym dictionaries are created in order to have a common language between multiple datasets. In the second step, we generated 6 features from circRNA-disease associations, circRNA-miRNA associations, disease-miRNA associations, circRNA sequence similarities, and disease semantic similarities. In the next step, selected circRNA and disease features are concatenated to have circRNA-disease pair representation. In the next step, we used a deep autoencoder (DAE) to extract high-level hidden features from fusion circRNA-disease pair representation. Finally, a deep neural network model (DNN) is trained with the final circRNA-disease pair vectorsand novel circRNA-disease pairs are predicted with the trained model. To find out optimal parameters in the methodology, different feature combinations, circRNA-disease pairs' embedding sizes, DAE layer sizes, and DNN layer sizes are tried in experiments.

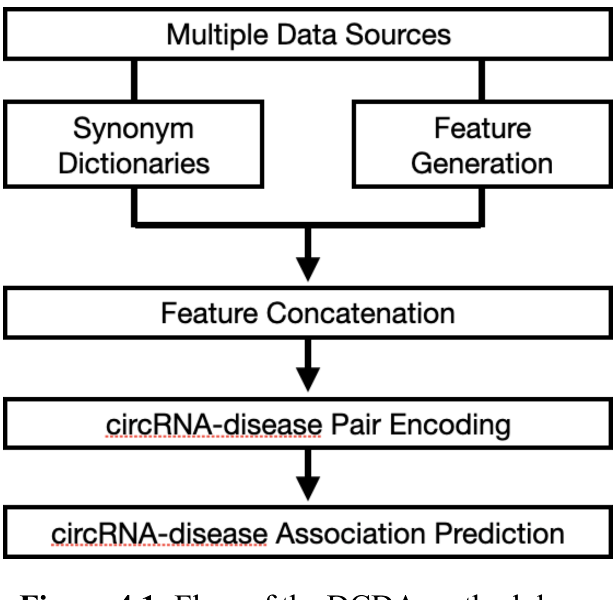


Figure 4.1: Flow of the DCDA methodology.

4.1. Data Preparation

In this section, dataset preparation steps and feature engineering methods will be described.

4.1.1. CircR2Disease: The Benchmark Dataset

CircR2Disease database is used as the benchmark dataset in our experiments. There are 739 entries which include 661 unique circRNAs and 100 diseases in the database. We selected only human disease-related circRNA-disease pairs in this study. In order to have a common language between all datasets used in the study, we mapped circRNA names using circRNA synonyms dictionary. Diseases names are also mapped using the disease synonyms dictionary. There was a small number of duplicated circRNA-disease pairs in the data, so we filtered out duplicated circRNA-disease pairs. Eventually, we ended up with 639 entries, 75 unique diseases, and 559 unique circRNAs. These confirmed circRNA-disease pairs are our positive sample. Since we do not have a dataset that shows not associated circRNA-disease pairs, we generated a negative sample set. In order to have a balanced dataset, we created the negative sample set such that it has the same number pairs as the positive set. We randomly created circRNA-disease pairs from circRNAs and diseases in the CircR2Disease database and selected pairs that are not confirmed. Finally, we have 639 positive samples and 639 negative samples, in total 1278 pairs, in our benchmark dataset.

4.1.2. Circ2Disease

We used the Circ2Disease dataset to see the performance of our methodology on a different dataset. 273 circRNAs-disease pairs are experimentally validated in the data. We changed circRNA and disease names by using our circRNA and disease synonyms dictionaries. After that, we excluded duplicated pairs in the data. Finally, we had 268 pairs, 233 unique circRNAs, and 60 unique diseases. We did the same steps in the previous section, we created our positive sample from these pairs and created a negative sample set with the same size as the positive sample set. In the end, we ended up with 536 pairs in the data that consists of 268 positive and 268 negative samples.

4.1.3. CircRNA Synonyms Dictionary

CircRNA synonyms are collected from CircR2Disease, circbase, Circ2Disease, circRNADisease, and CircFunBase databases. We looked at every circRNA-disease association information in the benchmark dataset and checked if the same circRNA-disease pair occurs in other datasets. We used PMID information to findcommon pairs. If there is a common circRNA-disease pair in another database, we collected the circRNA name or the circRNA id and put them in a dictionary as key, valuepairs. Key is the circRNA name mentioned in the CircR2Disease database and the valueis a set of synonym names of the circRNA mentioned in other datasets. From the circRNAsynonyms dictionary, 5 circRNA names and their synonyms are illustrated in Table 4.1.

Table 4.1: 5 circRNA names and their synonyms from circRNA synonym dictionary.

circRNA	**Synonyms**
hsa_circ_0000977	hsa_circ_001918, chr2:10784445-10808849
circ-Foxo3	hsa_circ_0006404, chr6:108984657-108986092
circZNF609	chr15:64791491-64792365, hsa_circ_0000615, hsa_circ_000193, cZNF609, cir-ZNF609
circ-FBXW7	novel_circ_022705
CircDOCK1	chr10:13701322-13717044, hsa_circRNA_004183, hsa_circ_100721, hsa_circ_0004183

4.1.4. Disease Synonyms Dictionary

Disease synonyms dictionary created using Circ2Disease, Circ2Disease, MeSH, and HMDD dataset. Each unique disease in those datasets are searched in the MeSH browser, if a disease found in the database websites redirects to the disease page. Disease page title is considered as the disease name and any search term (disease names from datasets) considered as synonyms of the disease. From the disease synonyms dictionary, 5 disease names and their synonyms are illustrated in Table 4.2.

Table 4.2: 5 disease names and their synonyms from disease synonym dictionary.

Disease	**Synonyms**
Lung Neoplasms	lung cancer
Diabetes, Gestational	Gestational Diabetes Mellitus
Urinary Bladder Neoplasms	Urinary Bladder Cancer, bladder urothelial carcinoma, bladder cancer, Bladder Neoplasms
Acne Vulgaris	acne
Tuberculosis	Mycobacterium tuberculosis Infection

4.1.5. Circular RNA Features

4.1.5.1. CircRNA-circRNA GIP kernel similarity (GIP_CD). According tothe CircR2Disease dataset, circRNA-disease association matrix *CD* is generated where rows are circRNAs and columns are diseases. Suppose an association between circRNA c_i and disease d_i, $CD(c_i, d_i)$ is set to 1, otherwise 0. We construct circRNA-circRNA GIPkernel similarity matrix from circRNA-disease associations (GIP_CD) using *CD*. The similarity between circRNA c_i and circRNA c_j is calculated as follows:

$$\text{GIP\_CD}(c_i, c_j) = exp(-\lambda \parallel CD(c_i) - CD(c_j) \parallel^2) \tag{4.1}$$

$$\lambda = \frac{1}{\frac{1}{nc}\sum_{i=0}^{nc} \| CD(c_i) \|^2} \tag{4.2}$$

Where $CD(c_i)$ and $CD(c_j)$ represents *i*th and *j*th rows in *CD*. λ is the regularization parameter which is used for controlling kernel bandwidth. *nc* stands for the number of unique circRNAs.

4.1.5.2. CircRNA-miRNA GIP kernel similarity (GIP_CM). We hypothesized that similar circRNAs have similar miRNA interactions and can be related to similar diseases. We used the circBank database since the circBank contains predicted human circRNA-miRNA relationships. We used entries related to the benchmark dataset's circRNAs and come up with 13624 entries with 841 unique miRNAs, and 48 unique diseases. Using the circBank database, circRNA-miRNA association matrix *CM* is generated where rows are circRNAs and columns are miRNAs. circRNA-miRNA GIP kernel similarity matrix (GIP_CM) is created from circRNA-miRNA association matrix *CM*. The similarity between circRNA c_i and circRNA c_j is calculated as follows:

$$\text{GIP\_CM}(c_i, c_j) = exp(-\lambda \parallel CM(c_i) - CM(c_j) \parallel^2) \tag{4.3}$$

$$\lambda = \frac{1}{\frac{1}{\mathrm{nc}}\sum_{\mathrm{i}=0}^{\mathrm{nc}} \| \mathrm{CD}(\mathrm{c_i}) \|^2} \tag{4.4}$$

Where $CM(c_i)$ and $CM(c_j)$ represents *i*th and *j*th rows in *CM*. λ is the regularization parameter which is used for controlling kernel bandwidth. *nc* stands for the number of unique circRNAs.

4.1.5.3. CircRNA-circRNA sequence similarity (SIM_CC). CircRNAs that have similar sequences are more likely to have similar functionalities and so related to similar diseases. According to this theorem, we calculated sequence similarity between circRNAs. circRNA sequence information is retrieved from the circBase database.We generated a circRNA-circRNA sequence similarity matrix called SIM_CC. The Levenstein distance algorithm is used to calculate the distance between circRNA sequences. The similarity between circRNA c_i and circRNA c_j is calculated as follows:

$$\text{SIM\_CC}(c_i, c_j) = 1 - \frac{lev\left(seq_{c_i}, seq_{c_j}\right)}{len\left(seq_{c_i}\right) + len\left(seq_{c_j}\right)} \tag{4.5}$$

Where seq_{c_i} is the sequence of circRNA c_i. *len* function calculates the length of the given sequence. *lev* function is the levenstein distance explained in 3.2. In the levenstein distance calculation, insertion, deletion and substitution costs set to 1, 1, and 2 respectively.

4.1.6. Disease Features

4.1.6.1. Disease-disease GIP kernel similarity (GIP_DC). Disease-diseaseGIP kernel similarity matrix (GIP_DC) is created using the CircR2Disease database. The similarity between disease d_i and disease d_j is calculated as follows:

$$\text{GIP\_DC}(d_i, d_j) = exp(-\lambda \parallel DC(d_i) - DC(d_j) \parallel^2) \tag{4.6}$$

$$\lambda = \frac{1}{\frac{1}{\mathrm{nd}} \sum_{\mathrm{i=0}}^{\mathrm{nd}} \| \mathrm{DC}(\mathrm{d_i}) \|^2} \tag{4.7}$$

DC is tranpose of *CD*. Where $DC(d_i)$ and $DC(d_j)$ represents *i*th and *j*th rows in *DC*. λ is the regularization parameter which is used for controlling kernel bandwidth. *nd* stands for the number of unique diseases.

4.1.6.2. Disease-miRNA GIP kernel similarity (GIP_DM). HMDD database contains experimentally supported human miRNA-disease relationships. In this study, we used HMDD v3.2. There are 35547 miRNA-disease association entries in this version, 1206 unique miRNA genes, and unique 893 diseases. We used entries related to the benchmark dataset's diseases and attained 13624 entries, 841 unique miRNAs, and 48 unique diseases. Using the HMDD database, disease-miRNA association matrix *DM* is generated where rows are diseases and columns are miRNAs. Disease-miRNA GIP kernel similarity matrix (GIP_DM) is created from disease-miRNA association matrix *DM*. The similarity between disease d_i and disease d_j is calculated as follows:

$$\text{GIP\_DM}(d_i, d_j) = exp(-\lambda \parallel DM(d_i) - DM(d_j) \parallel^2) \tag{4.8}$$

$$\lambda = \frac{1}{\frac{1}{\mathrm{nd}}\sum_{\mathrm{i=0}}^{\mathrm{nd}} \|\mathrm{DM}(\mathrm{d_i})\|^2} \tag{4.9}$$

Where $DM(d_i)$ and $DM(d_j)$ represents *i*th and *j*th rows in *DM*. λ is the regularization parameter which is used for controlling kernel bandwidth. *nd* stands for the number of unique diseases.

4.1.6.3. Disease-disease semantic similarity (SIM_DD). In the MeSH database, diseases have tree numbers that describe DAGs containing the disease. We searched unique diseases from CircR2Disease in the MeSH database and gathered tree numbers and entry terms of diseases. From 75 unique diseases from the benchmark dataset, we found 68 diseases in the MeSH database. As proposed in [11], we calculate the semantic similarity between disease d_i and disease d_j as follows: as follows:

$$\text{SIM\_DD}(d_i, d_j) = \frac{\sum_{e \in N_{d_i} \cap N_{d_j}} (D_{d_i}(e) + D_{d_j}(e))}{DV(d_i) + DV(d_j)} \tag{4.10}$$

$$DV(d) = \sum_{e \in N_d} D_d(e) \tag{4.11}$$

$$D_d(e) = -\log\left(\frac{num(DAGs(e))}{num(diseases)}\right) \tag{4.12}$$

Where *num*(*DAGs*(*e*)) represents the number of DAGs containing disease *e*, *num*(*diseases*) represents a number of all diseases. If there are no common trees between diseases, their similarity score is set to 0.

4.1.7. Feature Concatenation

circRNA-disease association fusion matrix (FUSION_MATRIX) is formed by concatenating feature vectors horizontally. For example, if we want to create a fusion vector that represents association between circRNA c_i and disease d_i using *GIP_CD*, *GIP_DC*, *GIP_DM*, and *SIM_DD* features, the fusion vector is formed by joining *GIP_CD*(c_i), *GIP_DC*(d_i), *GIP_DM*(d_i), and *SIM_DD*(d_i) vectors horizontally. Figure 4.2 shows how *FUSION_MATRIX*(c_i, d_i) is formed in the example scenario.

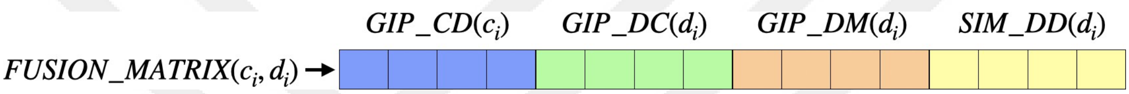


Figure 4.2: Feature concatenation of circRNA c_i and disease d_i using *GIP_CD*, *GIP_DC*, *GIP_DM*, and *SIM_DD* features.

4.2. Generating circRNA-disease Pair Embeddings

Autoencoders are a type of neural network that learns unique encodings of data in an unsupervised way. There are several studies which apply autoencoder neural networks to reveal unique hidden encodings in biological data [53, 54, 16, 17, 55]. In this study, we use a deep autoencoder (DAE) neural network as the feature extraction model. Autoencoders are composed of two symmetrical parts called encoder and decoder and an encoded input vector in the middle of the encoder and the decoder parts. The basic idea behind autoencoders is discovering a representation vector that can store all the critical information in a reduced size vector such that the encoded input vector can be used to create the original input in the output vector. Deep autoencoders have one or morehidden layers in both encoder and decoder parts. Figure 3.6 shows an example of a deep autoencoder.

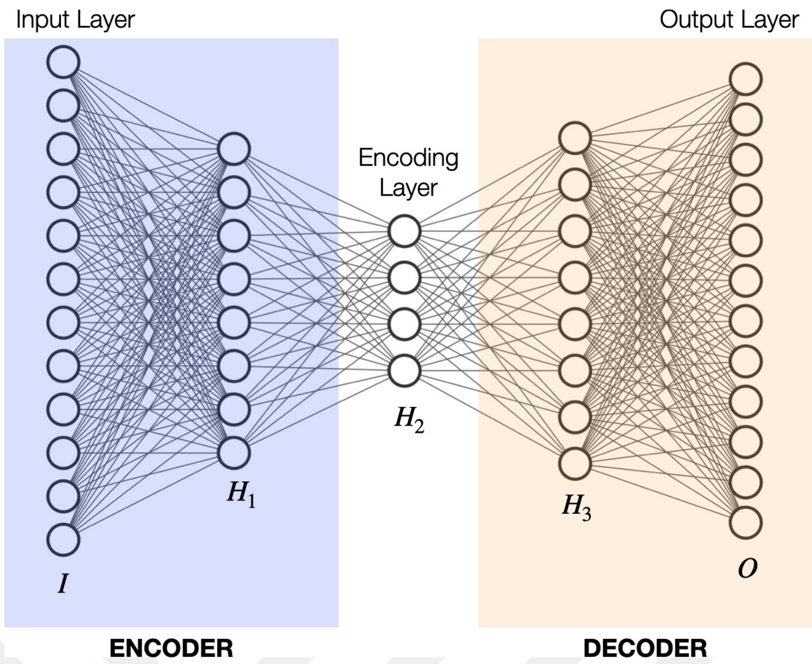


Figure 4.3: Deep autoencoder structure in our methodology.

In our methodology, a deep autoencoder with 3 hidden layers where 1 of them is the encoding layer is built. 4.3 shows the structure of our deep autoencoder model. Input and output layers are represented as I, O respectively. Hidden layers are represented as H_1, H_2, and H_3. Input, output, and hidden layer sizes are selected based on our experiments. The rule is that layer sizes of H_1 and H_3 are equal and proportional to input and encoding layer sizes. Layer size of H_1 and H_3 is calculated as follows:

$$size(H_1) = \frac{size(I)+size(H_2)}{2} \tag{4.13}$$

$$size(H_3) = size(H_1) \tag{4.14}$$

Where *size* function gets the layer size of the given layer. As an example, if an input layer and encoding layer sizes are 784 and 256 respectively, the layer sizes in the deep autoencoder are 784, 520, 256, 520, 784, respectively.

ReLU activation function is used in all layers. Adam optimizer is used as the optimization function.

4.3. Prediction of circRNA-disease Associations

In this study, a deep feed-forward deep neural network (DNN) model is used to predict circRNA-disease associations. The DNN model has one input layer, one output layer, and five hidden layers. All layers in the model are fully connected. ReLU activation function used in hidden layers and sigmoid function used in the output layer. Adam optimizer is used for optimization.

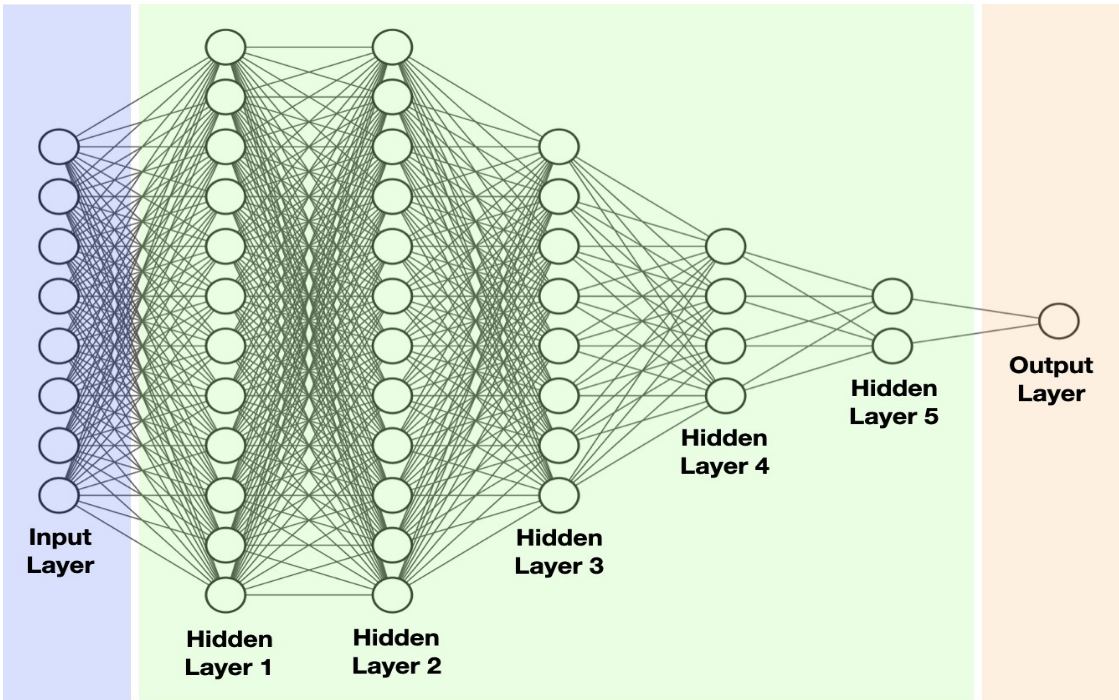


Figure 4.4: Deep Neural Network (DNN): circRNA-disease association prediction model.

5. EXPERIMENTS AND RESULTS

In this section, we run some experiments to find out the best inputs and parameters in our methodology. We also performed some experiments to measure the performance of our methodology and compare our methodology with state-of-the-art methodologies in the literature.

In all experiments, the following rules are applied:

- DAE structure presented in 4.3 is used, but layer sizes can vary according to the experiment,
- DNN structure presented in 4.4 is used, but layer sizes can vary according to the experiment,
- 5-fold cross-validation with stratified random sampling method is performed to have reliable results,
- AUC score is used as a performance metric for comparisons.

Measuring the performance of all possible feature combinations with a different set of model parameters is computationally expensive. That is why, we first tried to find the best feature combinations, and then find the best model parameters for the selected feature combination. Firstly, we performed a feature selection experiment and choose feature combinations that has the highest AUC scores. In the next steps, we used only selected feature combinations. In the model optimization part, we run our methodology with different DAE and DNN layer sizes. After these experiments, we selected our best feature combination, DAE layer sizes, and DNN layer sizes and showed the final methodology in the next part. Performance of the final methodology in detail in the next two parts. In the final part, we compared our methodology with state-of-the-art circRNA-disease association prediction methodologies.

5.1. Feature Selection

There are 3 circRNA features (*GIP_CD*, *GIP_CM*, *SIM_CC*) and 3 disease features (*GIP_DC*, *GIP_DM*, *SIM_DD*) generated from different sources. Using a lot of features does not always lead to having high model performance. Some features can cause noise effects in the model. Therefore, feature selection is an important step while building a model.

Table 5.1: 5-fold CV results on feature combinations.

CircRNA Features	Disease Features	AUC +- std. dev.
GIP_CD	**GIP_DC, SIM_DD**	**0.9764 +- 0.0047**
GIP_CD	**GIP_DC, GIP_DM, SIM_DD**	**0.9742 +- 0.0065**
GIP_CD	GIP_DM, SIM_DD	0.9667 +- 0.0132
GIP_CD	SIM_DD	0.9623 +- 0.0110
GIP_CD	GIP_DC	0.9594 +- 0.0175
GIP_CD	GIP_DC, GIP_DM	0.9581 +- 0.0137
GIP_CD, GIP_CM	GIP_DC, GIP_DM, SIM_DD	0.9565 +- 0.0165
GIP_CD, GIP_CM	GIP_DC, SIM_DD	0.9542 +- 0.0193
GIP_CD, SIM_CC	GIP_DM, SIM_DD	0.9524 +- 0.0069
GIP_CD, SIM_CC	GIP_DC, SIM_DD	0.9519 +- 0.0126
GIP_CD, GIP_CM	GIP_DM, SIM_DD	0.9516 +- 0.0214
GIP_CD, SIM_CC	GIP_DC, GIP_DM, SIM_DD	0.9463 +- 0.0161
GIP_CD, GIP_CM	GIP_DC, GIP_DM	0.9430 +- 0.0214
GIP_CD, SIM_CC	GIP_DC	0.9419 +- 0.0169
GIP_CD, SIM_CC	GIP_DC, GIP_DM	0.9385 +- 0.0047
GIP_CD, SIM_CC	SIM_DD	0.9364 +- 0.0088
GIP_CD, GIP_CM	SIM_DD	0.9329 +- 0.0247
GIP_CD, GIP_CM, SIM_CC	**GIP_DC, GIP_DM, SIM_DD**	**0.9288 +- 0.0180**
GIP_CD	GIP_DM	0.9280 +- 0.0071
GIP_CD, GIP_CM	GIP_DC	0.9211 +- 0.0385
GIP_CD, GIP_CM, SIM_CC	GIP_DC, SIM_DD	0.9151 +- 0.0195
GIP_CD, GIP_CM, SIM_CC	GIP_DM, SIM_DD	0.9108 +- 0.0236
GIP_CD, GIP_CM, SIM_CC	SIM_DD	0.9059 +- 0.0119
GIP_CD, SIM_CC	GIP_DM	0.9039 +- 0.0183
GIP_CD, GIP_CM, SIM_CC	GIP_DC, GIP_DM	0.8939 +- 0.0247
GIP_CD, GIP_CM, SIM_CC	GIP_DC	0.8791 +- 0.0266
GIP_CD, GIP_CM	GIP_DM	0.8748 +- 0.0252
GIP_CD, GIP_CM, SIM_CC	GIP_DM	0.8405 +- 0.0282

Table 5.2: 5-fold CV results on feature combinations (continued).

CircRNA Features	Disease Features	AUC +- std. dev.
GIP_CM, SIM_CC	GIP_DM, SIM_DD	0.7399 +- 0.0207
GIP_CM, SIM_CC	GIP_DC, SIM_DD	0.7393 +- 0.0326
SIM_CC	GIP_DC, SIM_DD	0.7354 +- 0.0191
SIM_CC	GIP_DM, SIM_DD	0.7319 +- 0.0319
GIP_CM, SIM_CC	SIM_DD	0.7265 +- 0.0182
SIM_CC	GIP_DC, GIP_DM, SIM_DD	0.7255 +- 0.0414
GIP_CM, SIM_CC	GIP_DC, GIP_DM, SIM_DD	0.7238 +- 0.0120
SIM_CC	GIP_DC, GIP_DM	0.7166 +- 0.0394
SIM_CC	SIM_DD	0.7151 +- 0.0335
GIP_CM, SIM_CC	GIP_DC, GIP_DM	0.6770 +- 0.0645
GIP_CM	GIP_DC, SIM_DD	0.6651 +- 0.0355
SIM_CC	GIP_DM	0.6647 +- 0.0469
GIP_CM, SIM_CC	GIP_DM	0.6247 +- 0.0485
GIP_CM	SIM_DD	0.6236 +- 0.0404
SIM_CC	GIP_DC	0.6075 +- 0.0956
GIP_CM, SIM_CC	GIP_DC	0.5944 +- 0.0758
GIP_CM	GIP_DC, GIP_DM	0.5847 +- 0.0500
GIP_CM	GIP_DC, GIP_DM, SIM_DD	0.5778 +- 0.0749
GIP_CM	GIP_DC	0.5757 +- 0.0238
GIP_CM	GIP_DM	0.5682 +- 0.0499
GIP_CM	GIP_DM, SIM_DD	0.5610 +- 0.0705

We created all combinations of circRNA and disease features and feed them to our methodology. We set encoding size to 128 and the DNN model's hidden layer sizes to 256, 512, 256, 128, and 64 respectively. We considered the AUC score as the performance score. Table 5.1 and 5.2 show circRNA and disease features used in the model and AUC score with standard deviation (std. dev.) of the model. As shown in the table, the model with all circRNA and disease features has AUC score of 0.9288 which is satisfying. However, some models with a fewer number of features have outstanding

AUC scores. We did not select only the best feature combination from the experiment because two combinations have really high scores and AUC scores are so close to each other. Therefore, we selected those two feature combinations that are highlighted in Table 5.1.

5.2. Model Optimization

Selecting the optimal embedding size of circRNA-disease pairs, and layer sizes in the prediction model is very critical in model optimization. Embedding size should be fair enough to represent circRNA-disease pairs such that the representation vector has a low vector size to run models efficiently and do not lose information. DNN prediction model also should have an appropriate structure to predict circRNA-disease associations successfully. There is no best model structure that works for every data and problem, so we need to perform experiments with different values to find out optimal parameters. In this section, we run our methodology with different encoding sizes and DNN hidden layer sizes. We used two feature combinations selected in Section 5.1 and considered AUC score as the comparison metric. We run the methodology with 4 different embedding sizes and 3 different hidden layer size combinations. As shown in Table 5.4, the best AUC score, 0.9794, is obtained by GIP_CD, GIP_DC, GIP_DM, and SIM_DD features with embedding size of 512 and DNN hidden layer size combination of 1024, 512, 256, 128, 64. Combining GIP_CD, GIP_DC, GIP_DM, and SIM_DD features generates a vector of 784 for a circRNA-disease pair. The autoencoder model generates the best model with an embedding size of 512 which means the autoencoder model finds hidden features from input features and reduced dimension in the same time.

5.3. Final Methodology

After selecting optimal features, embedding size, and DNN layer sizes we obtained the final methodology. Figure 5.1 shows all steps in the final methodology. In the following sections (5.4, 5.5, 5.6, 5.7, and 5.8) the proposed final methodology is used. We named our final methodology as DCDA.

Table 5.3: 5-fold CV results on different embedding sizes and DNN layer sizes.

CircRNA Features	Disease Features	Embdedding Size	DNN Layer Sizes	AUC +- std. dev.
GIP_CD	**GIP_DC, GIP_DM, SIM_DD**	**512**	**(1024, 512, 256, 128, 64)**	**0.9794 +- 0.0078**
GIP_CD	GIP_DC, GIP_DM, SIM_DD	1024	(256, 512, 256, 128, 64)	0.9792 +- 0.0089
GIP_CD	GIP_DC, GIP_DM, SIM_DD	1024	(1024, 512, 256, 128, 64)	0.9791 +- 0.0077
GIP_CD	GIP_DC, GIP_DM, SIM_DD	1024	(512, 512, 256, 128, 64)	0.9791 +- 0.0069
GIP_CD	GIP_DC, SIM_DD	1024	(512, 512, 256, 128, 64)	0.9784 +- 0.0109
GIP_CD	GIP_DC, GIP_DM, SIM_DD	512	(512, 512, 256, 128, 64)	0.9776 +- 0.0082
GIP_CD	GIP_DC, SIM_DD	1024	(256, 512, 256, 128, 64)	0.9773 +- 0.0022
GIP_CD	GIP_DC, SIM_DD	1024	(1024, 512, 256, 128, 64)	0.9765 +- 0.0087
GIP_CD	GIP_DC, SIM_DD	512	(1024, 512, 256, 128, 64)	0.9765 +- 0.0074
GIP_CD	GIP_DC, SIM_DD	256	(1024, 512, 256, 128, 64)	0.9764 +- 0.0064
GIP_CD	GIP_DC, SIM_DD	128	(256, 512, 256, 128, 64)	0.9764 +- 0.0047
GIP_CD	GIP_DC, SIM_DD	512	(256, 512, 256, 128, 64)	0.9762 +- 0.0077

Table 5.4: 5-fold CV results on different embedding sizes and DNN layer sizes (continued).

CircRNA Features	Disease Features	Embdedding Size	DNN Layer Sizes	AUC +- std. dev.
GIP_CD	GIP_DC, SIM_DD	512	(512, 512, 256, 128, 64)	0.9743 +- 0.0093
GIP_CD	GIP_DC, GIP_DM, SIM_DD	128	(256, 512, 256, 128, 64)	0.9742 +- 0.0065
GIP_CD	GIP_DC, SIM_DD	128	(1024, 512, 256, 128, 64)	0.9739 +- 0.0055
GIP_CD	GIP_DC, GIP_DM, SIM_DD	128	(512, 512, 256, 128, 64)	0.9736 +- 0.0073
GIP_CD	GIP_DC, GIP_DM, SIM_DD	256	(1024, 512, 256, 128, 64)	0.9735 +- 0.0132
GIP_CD	GIP_DC, SIM_DD	128	(512, 512, 256, 128, 64)	0.9735 +- 0.0049
GIP_CD	GIP_DC, SIM_DD	256	(512, 512, 256, 128, 64)	0.9731 +- 0.0051
GIP_CD	GIP_DC, GIP_DM, SIM_DD	512	(256, 512, 256, 128, 64)	0.9709 +- 0.0150
GIP_CD	GIP_DC, GIP_DM, SIM_DD	128	(1024, 512, 256, 128, 64)	0.9709 +- 0.0032
GIP_CD	GIP_DC, GIP_DM, SIM_DD	256	(512, 512, 256, 128, 64)	0.9701 +- 0.0126
GIP_CD	GIP_DC, GIP_DM, SIM_DD	256	(256, 512, 256, 128, 64)	0.9690 +- 0.0092
GIP_CD	GIP_DC, SIM_DD	256	(256, 512, 256, 128, 64)	0.9640 +- 0.0131

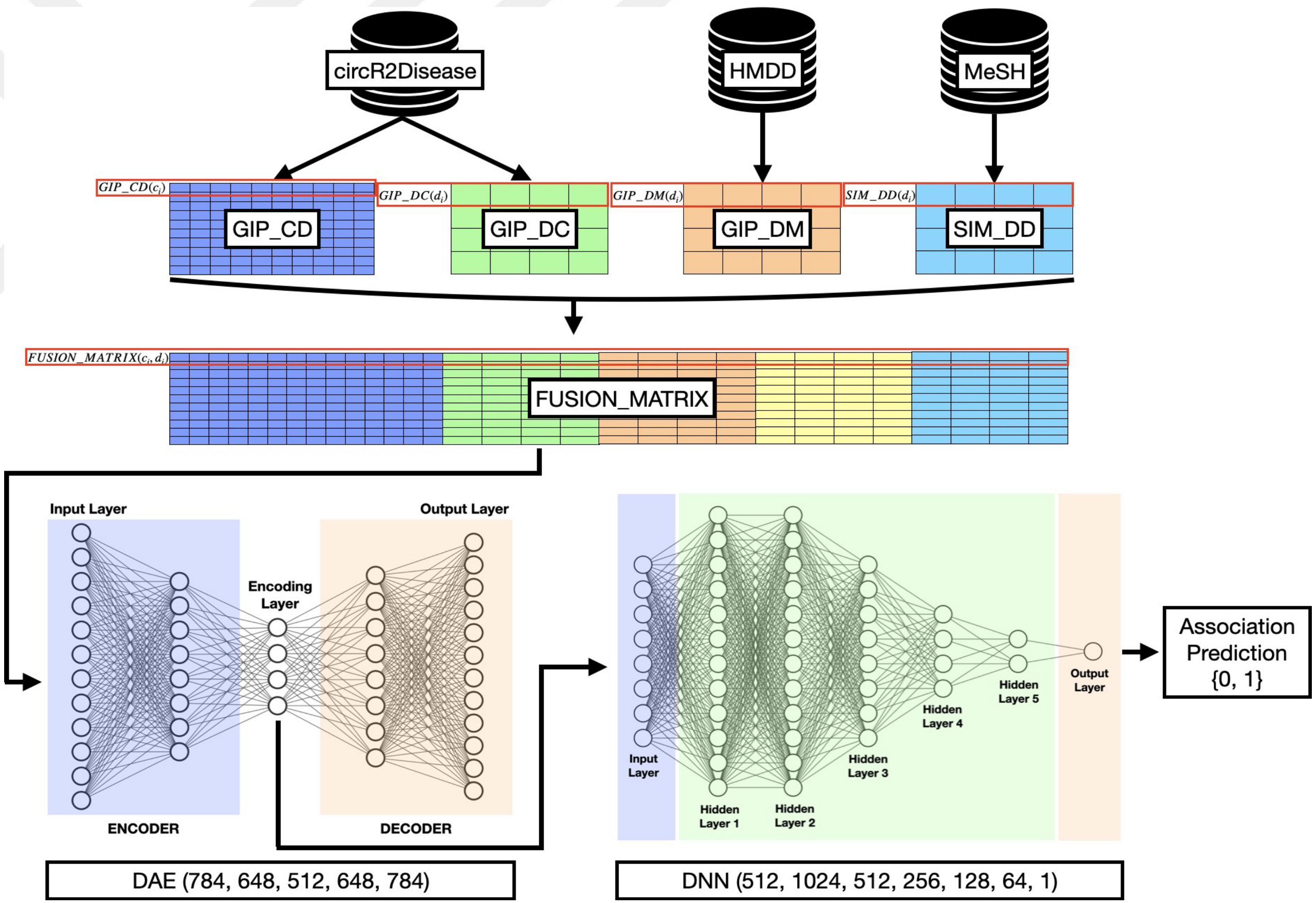


Figure 5.1: Flow of the DCDA methodology. GIP_CD, GIP_DC, GIP_DM, and SIM_DD features are created from circR2Disease, HMDD and MeSH databases. Then FUSION_MATRIX is generated by concatenating circRNA and disease features. Feature extraction is performed by the DAE model. Extracted features from the encoding layer of the DAE are fed to the DNN prediction model. Finally, DNN returns whether the circRNA-disease pair is associated (1) or not (0).

5.4. Comparison with Alternative Methods

In this part, we created 2 alternative methods to find out the contribution of the DAE model. In experiments, we used the same feature set described in the final methodology. Alternative methods are described below:

- Fusion-DNN: To see whether generating embedding vectors with deep autoencoder contributed to the model or not, we created a methodology same as DCDA but without the embedding generation part. Instead of running deep autoencoder to get embeddings, fusion vectors of circRNA-disease pairs are directly fed to the DNN prediction model.
- PCA-DNN: Another method is built which is same as DCDA but circRNA-disease pair embeddings are generated not with DAE but Principal Component Analysis (PCA) with the same encoding dimension used in DAE, 512.

Table 5.5: AUC scores of alternative methodologies on CircR2Disease dataset.

Method	DCDA	PCA-DNN	Fusion-DNN
AUC	0.9794	0.9713	0.9532

Table 5.5 shows results of alternative methodologies. As shown in the table, all alternative methods are successful at predicting circRNA-disease associations. This shows that we generated representative features for circRNAs and diseases. In addition to that, it is seen that using a deep autoencoder to get circRNA-disease pairs is very useful and increased the prediction capability.

5.5. Performance on the Benchmark Dataset

In order to evaluate DCDA's prediction performance, we performed 5-fold cross-validation on the benchmark dataset. We obtained 0.9241, 0.9262, 0.9013, 0.9531,and 0.9794 average scores of accuracy, F-1, precision, recall, and AUC respectively. Table 5.6 shows performance scores in each fold and the average score with standard deviation (std). Moreover, we calculated true positive rates and false-positive rates in each fold and plotted the ROC curves as shown in 5.2.

Table 5.6: 5-fold CV performance scores of the DCDA model on CircR2Disease dataset.

Fold	Accuracy	F-1	Precision	Recall	AUC
1	0.9180	0.9219	0.8794	0.9688	0.9758
2	0.9063	0.9091	0.8824	0.9375	0.9760
3	0.9141	0.9147	0.9077	0.9219	0.9686
4	0.9412	0.9438	0.9065	0.9844	0.9865
5	0.9412	0.9416	0.9308	0.9528	0.9900
Average	0.9241 +- 0.01	0.9262 +- 0.01	0.9013 +- 0.02	0.9531 +- 0.02	0.9794 +- 0.01

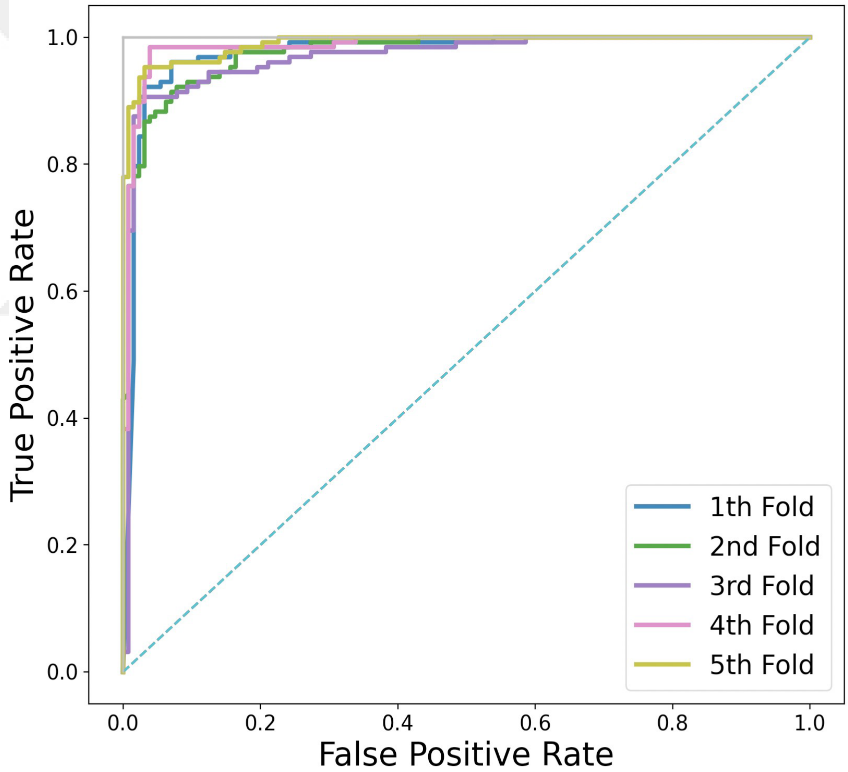


Figure 5.2: 5-fold CV ROC curves performed by DCDA model on CircR2Disease dataset.

5.6. Performance on Circ2Disease

To see the performance of the DCDA we run our methodology on a different dataset. We selected Circ2Disease dataset in this experiment. We performed 5-fold cross-validation on the data and obtained 0.8506, 0.8639, 0.8038, 0.9367, and 0.9425 average scores of accuracy, F-1, precision, recall, and AUC respectively. Table 5.7 shows performance scores in each fold and the average score with standard deviation (std). Figure 5.3 shows ROC curves of each fold.

Table 5.7: 5-fold CV performance scores of the DCDA model on Circ2Disease dataset.

Fold	Accuracy	F-1	Precision	Recall	AUC
1	0.9167	0.9189	0.8947	0.9444	0.9688
2	0.8224	0.8348	0.7869	0.8889	0.9203
3	0.7757	0.8095	0.7083	0.9444	0.9261
4	0.8411	0.8522	0.7903	0.9245	0.9345
5	0.8972	0.9043	0.8387	0.9811	0.9630
Average	0.8506 +- 0.05	0.8639 +- 0.04	0.8038 +- 0.06	0.9367 +- 0.03	0.9425 +- 0.02

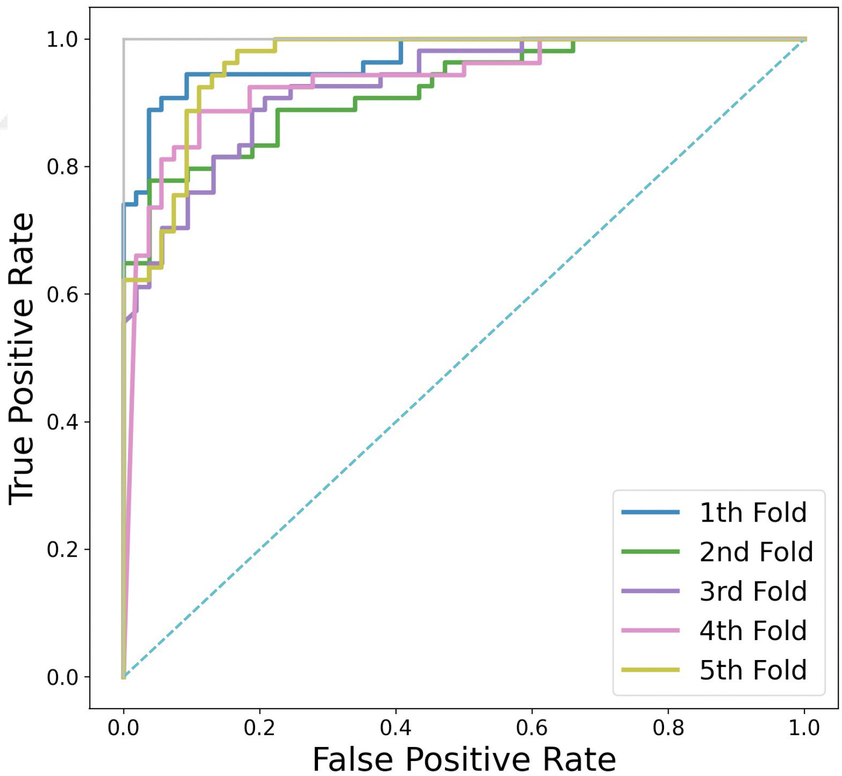


Figure 5.3: 5-fold CV ROC curves performed by DCDA model on Circ2Disease dataset.

5.7. Comparison of DCDA With Relative Methodologies

We compared DCDA model's performance with five state-of-the-art studies including MSFCNN[12], AE-RF[17], AE-DNN[16], iCircDA-MF[18], KATZHCDA[10].

In order to make a reliable comparison, we compared studies based on 5-fold CV results on the same dataset, CircR2Disease. All compared methodologies used only human-related circRNA-disease associations but applied different filtering processes. Unlike DCDA, these five studies did not use circRNA or disease synonyms to have a consistent dataset. MSFCNN, AE-RF, and AE-DNN are based on machine learning prediction models, KATZHCDA is a graph-based prediction model, and iCircDA-MF's prediction model is based on matrix-factorization. Like in our methodology, AE-DNN and AE-RF use an autoencoder model for feature extraction. As shown in Table 5.8, DCDA is superior to the other state-of-the-art methods with an AUC score of 0.9794.

Table 5.8: AUC scores of different circRNA-disease association prediction methodologies on CircR2Disease dataset.

Method	DCDA	MSFCNN[1]	AE-RF[2]	AE-DNN[3]	iCircDA-MF[4]	KATZHCDA[5]
AUC	0.9794	0.9525	0.9486	0.9392	0.9178	0.7936

[1] Results obtained by the model reported in [12]

[2] Results obtained by the model reported in [17]

[3] Results obtained by the model reported in [16]

[4] Results obtained by the model reported in [18] with parameter $k = 2$, $r = 70$, $\alpha = 2 \times 10 - 3$, $\beta = 1 \times 10 - 3$

[5] Results obtained by the model reported in [10]

5.8. Novel circRNA-disease Associations

To verify the prediction performance of the DCDA methodology, we investigated the novel circRNA-disease pairs predicted by the DCDA methodology. For training, the benchmark dataset is used to train DAE and DNN. Then, we created a test set by taking all possible combinations between unique circRNAs and unique diseases in the benchmark dataset and excluded validated pairs. As a result, we had 41286 pairs.

Prediction results are sorted by the prediction score of circRNA-disease pairs and the highest top-10 novel associations are examined. If the predicted association is found in the literature, the PMID of the study is listed in the table. As shown in Table

5.9, 7 out of 10 novel circRNA-disease associations are confirmed in publications. For example, novel associations between circRNA CDR1as with breast neoplasms, urinary bladder neoplasms, and glioma diseases are predicted by our methodology and researches confirmed these associations.

Table 5.9: Top 10 novel circRNA-disease association predictions by DCDA methodology.

Rank	**Disease**	**circRNA**	**Evidence***
1	Breast Neoplasms	CDR1as	31245927
2	Urinary Bladder Neoplasms	CDR1as	29694981
3	Glioma	CDR1as	26683098
4	Glioma	circBRAF	33650075
5	Glioma	Cir-ITCH	29887952
6	Esophageal Neoplasms	Cir-ITCH	25749389
7	Carcinoma, Pancreatic Ductal	hsa_circ_0001649	29969694
8	Carcinoma, Basal Cell	hsa_circ_0000284	Unconfirmed
9	Urinary Bladder Neoplasms	hsa_circ_0001649	Unconfirmed
10	Colorectal Neoplasms	hsa_circ_0067934	Unconfirmed

*The PMID of literature in PubMed that are evidence of the corresponding circRNA-disease association.

DCDA is also evaluated for discovering potential circRNAs for three important neoplasm types: lung neoplasms, breast neoplasms, and urinary bladder neoplasms. Lung neoplasm is the most fatal common cancer type around the world. The number of cancer patients who suffer from lung neoplasms is increasing and causes deaths over decades [56]. The second most common cancer type is breast cancer and it is the most frequent neoplasm among women [57]. Bladder neoplasms is another frequently seen cancer type and each year approximately 550,000 patients are diagnosed [58]. We looked closely at unknown circRNA-disease association predictions for these cancer types and examined the top 10 circRNAs that the model predicted.

Table 5.10: Top 10 predicted circRNAs for Lung Neoplasms.

Rank	circRNA	Evidence*
1	hsa_circ_0000284	32550825
2	CDR1as	31917898
3	circRHOBTB3	33163392
4	circARHGAP5	Unconfirmed
5	circSMARC1A	Unconfirmed
6	circFAT1	33884267
7	hsa_circRNA_103076	Unconfirmed
8	hsa_circRNA_103390	Unconfirmed
9	circRNA_101877	Unconfirmed
10	circ-LDLRAD3	32658600

*The PMID of literature in PubMed that are evidence of the corresponding circRNA-disease association.

Table 5.11: Top 10 predicted circRNAs for Breast Neoplasms.

Rank	circRNA	Evidence*
1	CDR1as	31245927
2	hsa_circRNA_101113	Unconfirmed
3	hsa_circRNA_104692	Unconfirmed
4	hsa_circRNA_103801	Unconfirmed
5	hsa_circRNA_103309	Unconfirmed
6	hsa_circRNA_102485	Unconfirmed
7	hsa_circRNA_100241	Unconfirmed
8	circ-NT5C2	Unconfirmed
9	cir-GLI2	Unconfirmed
10	hsa_circ_0001564	Unconfirmed

*The PMID of literature in PubMed that are evidence of the corresponding circRNA-disease association.

Table 5.12: Top 10 predicted circRNAs for Urinary Bladder Neoplasms.

Rank	**circRNA**	**Evidence***
1	CDR1as	29694981
2	hsa_circ_0001649	Unconfirmed
3	circPVT1	Unconfirmed
4	hsa_circ_0001313	Unconfirmed
5	circZNF609	Unconfirmed
6	hsa_circRNA_102619	Unconfirmed
7	circRNA0047905	Unconfirmed
8	hsa_circRNA_000167	Unconfirmed
9	circSMARCA5	35116915
10	circ-Foxo3	31802888

*The PMID of literature in PubMed that are evidence of the corresponding circRNA-disease association.

6. CONCLUSIONS AND FUTURE WORK

In recent years, the importance of circRNAs in diagnoses of diseases and disease treatments come to light. Improvement of prediction algorithms, especiallydeep learning models, stimulated using machine learning algorithms while discovering disease-related circRNAs. In this thesis, deep learning-based circRNA-disease association prediction methodology, called DCDA, is presented. We generated circRNA and disease feature vectors using different sources of information such as circRNA-miRNA associations, disease-miRNA associations, circRNA sequence similarity, and disease semantic similarities. We created circRNA and disease synonym dictionaries to have common language between various datasets. We generated several feature vectors for circRNAs and diseases to find out optimal features. As a consequence of feature selection experimentation, features derived from known circRNA-disease interactions anddisease-miRNA associations are selected to generate circRNA and disease feature vectors. We represented circRNA-disease pairs by concatenating circRNA and disease feature vectors and generated representative pair vectors in lower-dimensional space through a deep autoencoder. Finally, we build a deep neural network to predict whether the pair is associated or not.

To identify the predictive performance of our methodology, we performed several experiments. We tested DCDA on the CircR2Disease dataset, which is a widely-used dataset in the literature, and showed that the performance of DCDA has an undeniable performance with an AUC score of 0.9794. We also tested DCDA on another widely-used dataset, Circ2Disease, and achieved an AUC score of 0.9425. Besides these experimentations, we compared our methodology with state-of-the-art methodologies in the literature. Our model outperformed the top-ranking methodology, MSFCNN, with a 2.82% AUC score.

Relying on DCDA’s prospering performance in all experimentations, we looked at what DCDA predicts about unknown circRNA-disease pairs. When we investigated circRNA-disease pairs that have the highest predictive association score, we saw that 7 out of the top-10 pair associations are confirmed in the literature. We also investigated predicted circRNAs for 3 important cancers types (lung, breast, and urinary bladder neoplasms), and saw that some of the circRNAs are confirmed in the literature recently.

There are multiple benefits of this research. First of all, circRNA-disease pair embeddings generated by the deep autoencoder model can be used in other related studies. Relying on the successful performance of DCDA, unconfirmed predicted circRNA-disease pairs are worth being analyzed and examined by biologists. Starting examination from most likely associated circRNAs will bring forth saving time and spending resources on unlikely associated circRNA-disease pairs.

6.1. Future Work

In further study, various other strategies can be applied to such research. Additional data sources could be used to extract more circRNA and disease features such as circRNA-gene associations, and disease-disease similarities based on disease symptoms. Since deep learning algorithms work better with big data, training data could be expanded with recently found circRNA-disease interactions. Moreover, parameters of the deep autoencoder and the deep neural network model architectures can be optimized with the help of computational power.

REFERENCES

[1] Lesca M Holdt, Alexander Kohlmaier, and Daniel Teupser. “Molecular roles and function of circular RNAs in eukaryotic cells”. In: *Cellular and Molecular Life Sciences* 75.6 (2018), pp. 1071–1098.

[2] Heinz L Sanger et al. “Viroids are single-stranded covalently closed circular RNA molecules existing as highly base-paired rod-like structures”. In: *Proceedings of the National Academy of Sciences* 73.11 (1976), pp. 3852–3856.

[3] Ming-Ta Hsu and Miguel Coca-Prados. “Electron microscopic evidence for the circular form of RNA in the cytoplasm of eukaryotic cells”. In: *Nature* 280.5720 (1979), pp. 339–340.

[4] Minghon Lu. “Circular RNA: Functions, applications and prospects”. In: *ExRNA* 2.1 (2020), pp. 1–7.

[5] Xiyong Wang et al. “Increased circular RNA hsa_circ_0012673 acts as a sponge of miR-22 to promote lung adenocarcinoma proliferation”. In: *Biochemical and biophysical research communications* 496.4 (2018), pp. 1069–1075.

[6] Chunyan Fan et al. “CircR2Disease: a manually curated database for experimentally supported circular RNAs associated with various diseases”. In: *Database* 2018 (2018).

[7] Mercy Rophina et al. “Circad: a comprehensive manually curated resource of circular RNA associated with diseases”. In: *Database* 2020 (2020).

[8] Dongxia Yao et al. “Circ2Disease: a manually curated database of experimentally validated circRNAs in human disease”. In: *Scientific reports* 8.1 (2018), pp. 1–6.

[9] Xianwen Meng et al. “CircFunBase: a database for functional circular RNAs”. In: *Database* 2019 (2019).

[10] Chunyan Fan, Xiujuan Lei, and Fang-Xiang Wu. “Prediction of CircRNA-disease associations using KATZ model based on heterogeneous networks”. In: *International journal of biological sciences* 14.14 (2018), p. 1950.

[11] Lei Wang et al. “An efficient approach based on multi-sources information to predict circRNA–disease associations using deep convolutional neural network”. In: *Bioinformatics* 36.13 (2020), pp. 4038–4046.

[12] Chunyan Fan, Xiujuan Lei, and Yi Pan. “Prioritizing CircRNA-disease associations with convolutional neural network based on multiple similarity feature fusion”. In: *Frontiers in Genetics* 11 (2020), p. 1042.

[13] Lei Wang et al. “Predicting circRNA-disease associations using deep generative adversarial network based on multi-source fusion information”. In: *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2019, pp. 145–152.

[14] Kai Zheng et al. “iCDA-CGR: Identification of circRNA-disease associations based on Chaos Game Representation”. In: *PLOS Computational Biology* 16.5 (2020), e1007872.

[15] Erxia Ge et al. “Predicting human disease-associated circRNAs based on locality-constrained linear coding”. In: *Genomics* 112.2 (2020), pp. 1335–1342.

[16] K Deepthi and AS Jereesh. “An ensemble approach for CircRNA-disease association prediction based on autoencoder and deep neural network”. In: *Gene* 762 (2020), p. 145040.

[17] K Deepthi and AS Jereesh. “Inferring potential CircRNA–disease associations via deep autoencoder-based classification”. In: *Molecular Diagnosis & Therapy* 25.1 (2021), pp. 87–97.

[18] Hang Wei and Bin Liu. “iCircDA-MF: identification of circRNA-disease associations based on matrix factorization”. In: *Briefings in bioinformatics* 21.4 (2020), pp. 1356–1367.

[19] Lei Wang et al. “GCNCDA: A new method for predicting circRNA-disease associations based on Graph Convolutional Network Algorithm”. In: *PLOS Computational Biology* 16.5 (2020), e1007568.

[20] Kuei-Yang Hsiao, H Sunny Sun, and Shaw-Jenq Tsai. “Circular RNA–new member of noncoding RNA with novel functions”. In: *Experimental Biology and Medicine* 242.11 (2017), pp. 1136–1141.

[21] Peter G Zaphiropoulos. “Circular RNAs from transcripts of the rat cytochrome P450 2C24 gene: correlation with exon skipping”. In: *Proceedings of the National Academy of Sciences* 93.13 (1996), pp. 6536–6541.

[22] Christin E Burd et al. “Expression of linear and novel circular forms of an INK4/ARF-associated non-coding RNA correlates with atherosclerosis risk”. In: *PLoS Genet* 6.12 (2010), e1001233.

[23] Agus Surono et al. “Circular dystrophin RNAs consisting of exons that were skipped by alternative splicing”. In: *Human molecular genetics* 8.3 (1999), pp. 493–500.

[24] Xianwen Meng et al. “Circular RNA: an emerging key player in RNA world”. In: *Briefings in bioinformatics* 18.4 (2017), pp. 547–557.

[25] Julia Salzman et al. “Circular RNAs are the predominant transcript isoform from hundreds of human genes in diverse cell types”. In: *PloS one* 7.2 (2012), e30733.

[26] Sebastian Memczak et al. “Circular RNAs are a large class of animal RNAs with regulatory potency”. In: *Nature* 495.7441 (2013), pp. 333–338.

[27] Yang Zhang et al. “Circular intronic long noncoding RNAs”. In: *Molecular cell* 51.6 (2013), pp. 792–806.

[28] Zhaoyong Li et al. “Exon-intron circular RNAs regulate transcription in the nucleus”. In: *Nature structural & molecular biology* 22.3 (2015), p. 256.

[29] William R Jeck et al. “Circular RNAs are abundant, conserved, and associated with ALU repeats”. In: *Rna* 19.2 (2013), pp. 141–157.

[30] Zhipeng Lu et al. “Metazoan tRNA introns generate stable circular RNAs in vivo”. In: *Rna* 21.9 (2015), pp. 1554–1565.

[31] Yibing Yang et al. “Novel role of FBXW7 circular RNA in repressing glioma tumorigenesis”. In: *JNCI: Journal of the National Cancer Institute* 110.3 (2018), pp. 304–315.

[32] Hai-Feng Liang et al. “Circular RNA circ-ABCB10 promotes breast cancer proliferation and progression through sponging miR-1271”. In: *American journal of cancer research* 7.7 (2017), p. 1566.

[33] Ming Liu et al. “Circbank: a comprehensive database for circRNA with standard nomenclature”. In: *RNA biology* 16.7 (2019), pp. 899–905.

[34] Petar Glažar, Panagiotis Papavasileiou, and Nikolaus Rajewsky. “circBase: a database for circular RNAs”. In: *Rna* 20.11 (2014), pp. 1666–1670.

[35] Carolyn E Lipscomb. “Medical subject headings (MeSH)”. In: *Bulletin of the Medical Library Association* 88.3 (2000), p. 265.

[36] Zhou Huang et al. “HMDD v3. 0: a database for experimentally supported human microRNA–disease associations”. In: *Nucleic acids research* 47.D1 (2019), pp. D1013–D1017.

[37] Twan Van Laarhoven, Sander B Nabuurs, and Elena Marchiori. “Gaussian interaction profile kernels for predicting drug–target interaction”. In: *Bioinformatics* 27.21 (2011), pp. 3036–3043.

[38] Sergey Grashchenko. *Levenshtein Distance Computation*. [Online; accessed January 31, 2022]. 2021. URL: `https://www.baeldung.com/cs/levenshtein-distance-computation`.

[39] James Z Wang et al. “A new method to measure the semantic similarity of GO terms”. In: *Bioinformatics* 23.10 (2007), pp. 1274–1281.

[40] Menglu Li et al. “Prediction of circRNA-disease associations based on inductive matrix completion”. In: *BMC medical genomics* 13.5 (2020), pp. 1–13.

[41] Buse Kara. *PERCEPTRON NEDİR ?* [Online; accessed January 31, 2022]. 2019. URL: `https://medium.com/@buse.kara7998/perceptron-nedi%CC%87r-a15a7c8e5276`.

[42] Marvin Minsky and Seymour Papert. “An introduction to computational geometry”. In: *Cambridge tiass., HIT* (1969).

[43] IBM Cloud Education. *Neural Networks*. [Online; accessed January 31, 2022]. 2020. URL: https://www.ibm.com/cloud/learn/neural-networks.

[44] Shruti Jadon. *Introduction to Different Activation Functions for Deep Learning*. [Online; accessed January 31, 2022]. 2018. URL: https://medium.com/@shrutijadon10104776/survey-on-activation-functions-for-deep-learning-9689331ba092.

[45] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[46] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. *Learning internal representations by error propagation*. Tech. rep. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[47] Pierre Baldi. "Autoencoders, unsupervised learning, and deep architectures". In: *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings. 2012, pp. 37–49.

[48] Arden Dertat. *Applied Deep Learning - Part 3: Autoencoders*. [Online; accessed January 31, 2022]. 2017. URL: https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798.

[49] *Train-test split and cross-validation*. [Online; accessed January 31, 2022]. URL: https://digitalmind.io/post/train-test-split-and-cross-validation#train-test-split-and-cross-validation.

[50] Pieter G de Vries. "Stratified random sampling". In: *Sampling theory for forest inventory*. Springer, 1986, pp. 31–55.

[51] James A Hanley and Barbara J McNeil. "The meaning and use of the area under a receiver operating characteristic (ROC) curve." In: *Radiology* 143.1 (1982), pp. 29–36.

[52] Jerome Fan, Suneel Upadhye, and Andrew Worster. "Understanding receiver operating characteristic (ROC) curves". In: *Canadian Journal of Emergency Medicine* 8.1 (2006), pp. 19–20.

[53] Lujia Chen et al. "Learning a hierarchical representation of the yeast transcriptomic machinery using an autoencoder model". In: *BMC bioinformatics*. Vol. 17. 1. BioMed Central. 2016, pp. 97–107.

[54] Davide Chicco, Peter Sadowski, and Pierre Baldi. "Deep autoencoder neural networks for gene ontology annotation predictions". In: *Proceedings of the 5th ACM conference on bioinformatics, computational biology, and health informatics*. 2014, pp. 533–540.

[55] Jie Tan et al. "Adage-based integration of publicly available pseudomonas aeruginosa gene expression data with denoising autoencoders illuminatesmicrobe-host interactions". In: *MSystems* 1.1 (2016).

[56] Kwun M Fong, Tobias Welte, et al. *World Lung Day: what, why, and where to?* 2020.

[57] Caroline Tanure Castro et al. "Relationship between searching for breast cancer on the web and number of cases of breast cancer". In: *Brazilian Journal of Oncology* 17 (2021).

[58] Anke Richters, Katja KH Aben, and Lambertus ALM Kiemeney. "The global burden of urinary bladder cancer: an update". In: *World journal of urology* 38.8 (2020), pp. 1895–1904.