Mehmet Emin ORHAN

A M.Sc. Thesis

AGU 2024

ADVANCING MACHINE LEARNING ANALYSIS OF NON-CODING RNA: A NOVEL APPROACH OF NEGATIVE SEQUENCE GENERATION

M.Sc. THESIS
SUBMITTED TO THE DEPARTMENT OF BIOENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER

By
Mehmet Emin ORHAN
May 2024

ADVANCING MACHINE LEARNING ANALYSIS OF NON-CODING RNA: A NOVEL APPROACH OF NEGATIVE SEQUENCE GENERATION

A THESIS
SUBMITTED TO THE DEPARTMENT OF BIOENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF
ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Mehmet Emin ORHAN
May 2024

SCIENTIFIC ETHICS COMPLIANCE

I hereby declare that all information in this document has been obtained 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 materials and results that are not original to this work.

Name-Surname: Mehmet Emin ORHAN
Signature :

REGULATORY COMPLIANCE

M.Sc. thesis titled Advancing Machine Learning Analysis of Non-Coding RNA: A Novel Approach of Negative Sequence Generation has been prepared in accordance with the Thesis Writing Guidelines of the Abdullah Gül University, Graduate School of Engineering & Science.

Prepared By	Advisor
Mehmet Emin ORHAN	Assoc. Prof. Dr. Müşerref Duygu SAÇAR DEMİRCİ
Signature	Signature

Head of the Bioengineering Program
Assist. Prof. Dr. Emel Başak GENCER AKÇOK
Signature

ACCEPTANCE AND APPROVAL

M.Sc thesis titled Advancing Machine Learning Analysis of Non-Coding RNA: A Novel Approach of Negative Sequence Generation and prepared by Mehmet Emin Orhan has been accepted by the jury in the Bioengineering Graduate Program at Abdullah Gül University, Graduate School of Engineering & Science.

24 /05/2024

(Thesis Defense Exam Date)

JURY:

Advisor : Assoc. Prof. Dr. Müşerref Duygu SAÇAR DEMİRCİ

Member : Prof. Dr. Jens ALLMER

Member : Assist. Prof. Dr. Emel Başak GENCER AKÇOK

APPROVAL:

The acceptance of this M.Sc. thesis has been approved by the decision of the Abdullah Gül University, Graduate School of Engineering & Science, Executive Board dated ….. /….. / …….. and numbered ……..……. .

…….. /…….. / ……..

(Date)

Graduate School Dean
Prof. İrfan ALAN

ABSTRACT

IN SILICO ANALYSIS OF RNA INTERACTIONS

Mehmet Emin ORHAN

MSc. in Bioengineering

Advisor: Assoc. Prof. Müşerref Duygu SAÇAR DEMİRCİ

April 2024

Many supervised machine learning models have been developed for the classification and identification of non-coding RNA (ncRNA) sequences. These models play a significant role in the diagnosis and treatment of various diseases. During such analyses, positive learning datasets typically consist of known ncRNA examples, some of which may even be confirmed with strong experimental evidence. However, there is no database of validated negative sequences for ncRNA classes or standardized methodologies for generating high quality negative samples. To overcome this challenge, a new method for generating negative data called the NeRNA (Negative RNA) method has been developed in this study. NeRNA generates negative sequences using known ncRNA sequences and their octal representations, similar with frame shift mutations found in biology but without base deletions or insertions. In this thesis, the NeRNA method was tested separately with four different ncRNA datasets, including microRNA (miRNA), transfer RNA (tRNA), long non-coding RNA (lncRNA), and circular RNA (circRNA). Additionally, a species-specific case study was conducted to demonstrate and compare the performance of the study's miRNA predictions. The results of 1000-fold cross-validation on machine learning algorithms such as Decision Trees, Naive Bayes, Random Forest classifiers, and deep learning algorithms like Multilayer Perceptrons, Convolutional Neural Networks, and Simple Feedforward Neural Networks showed that models developed using datasets generated by NeRNA exhibited significantly high prediction performance. NeRNA has been published as an easy-to-use, updatable, and modifiable KNIME workflow, along with example datasets and required extensions that can be downloaded and utilized. NeRNA is designed specifically as a powerful tool for RNA sequence data analysis.

Keywords: Machine Learning, Non-coding RNA, Data Generation, Negative Data

ÖZET

RNA ETKİLEŞİMLERİNİN İN SİLİCO ANALİZİ

Mehmet Emin ORHAN

Biyomühendislik Anabilim Dalı Yüksek Lisans

Tez Danışmanı: Doç.Dr. Müşerref Duygu SAÇAR DEMİRCİ

Nisan 2024

Kodlanmayan RNA (ncRNA) dizilerinin sınıflandırılması tanımlanması için birçok denetimli makine öğrenimi modelleri geliştirilmiştir. Bu modeller birçok hastalığın tanı ve tedavisinde önemli rol oynamaktadır. Bu tür analizler sırasında, pozitif öğrenme veri kümeleri genellikle bilinen ncRNA örneklerinden oluşur ve hatta bazıları güçlü deneysel verilerle doğrulanmış olabilir. Buna karşılık, ncRNA sınıfları için doğrulanmış negatif dizileri içeren bir veri tabanı veya yüksek kaliteli negatif örnek oluşturmayı sağlayan standart metodolojiler bulunmamaktadır. Bu zorluğun üstesinden gelebilmek için, bu çalışmada yeni bir negatif veri oluşturma yöntemi olan NeRNA (negatif RNA) yöntemi geliştirilmiştir. NeRNA, bilinen ncRNA dizilerini ve sekizli gösterim yapılarını kullanılarak negatif diziler oluşturur, bu oluşturma biyoloji de bulunan çerçeve kayması mutasyonlarına benzer bir şekilde ancak baz silme veya ekleme olmadan gerçekleşir. Bu tez kapsamında, mikroRNA (miRNA), transfer RNA (tRNA), uzun kodlamayan RNA (lncRNA) ve dairesel RNA (circRNA) dahil olmak üzere dört farklı ncRNA veri kümesi ile ayrı ayrı test edilmiştir. Ayrıca, çalışmanın miRNA tahminleri üzerinde performansını göstermek ve karşılaştırmak için türe özgü bir vaka analizi gerçekleştirilmiştir. Çalışma boyunca kullanılan Karar Ağacı, Naieve Bayes, Rastgele Orman sınıflandırıcıları gibi makine öğrenimi algoritmaları ve Çok Katmanlı Algılayıcı, Evrişimli Sinir Ağı ve Basit İleri Beslemeli Sinir Ağları gibi derin öğrenme algoritmaları üzerinde yapılan 1000 kat çapraz doğrulama sonuçları, NeRNA tarafından oluşturulan veri kümeleri kullanılarak elde edilen modellerin önemli ölçüde yüksek tahmin performansı sağladığını göstermektedir. NeRNA, örnek veri kümeleri ve gerekli uzantılarla birlikte indirilebilen, kullanımı kolay, güncellenebilir ve değiştirilebilir bir KNIME iş akışı olarak yayınlanmaktadır. NeRNA, özellikle RNA dizisi veri analizi için güçlü bir araç olarak tasarlanmıştır.

Anahtar kelimeler: Makine Öğrenimi, Kodlanmayan RNA, Veri Üretimi, Negatif Veri

Acknowledgements

I would like to express my deepest gratitude to Assoc. Prof. Dr. Müşerref Duygu SAÇAR DEMİRCİ, whose invaluable guidance and support have been instrumental throughout my master's degree journey, including the preparation of this thesis. Her patient and encouraging approach have shaped my academic career, and I will always be grateful to her. I am also thankful to Assoc. Prof. Dr. Yılmaz Mehmet DEMİRCİ, for his unwavering support and presence. Lastly, I extend my heartfelt thanks to their sweet daughter, Ada.

I am especially grateful to Asst. Prof. Dr. Sebiha ÇEVİK KAPLAN and Asst. Prof. Dr. Oktay İsmail KAPLAN for their invaluable support and guidance throughout my M.Sc. journey.

I extend my heartfelt gratitude to my dear friend Emre Can ÇİFTÇİ for always being there for me and to my dear sister Şeyma ŞAKAR, who has always been with me and supported me with her perspectives and guided me, and to my brother Arda ÜZMEZ for standing by me through both good and challenging times. I also want to commemorate my dear friend Ali Haydar DİNÇALP with this thesis. You are always in our hearts.

I extend my deepest gratitude to my father, who has been a guiding light and a pillar of support throughout my life, and to my beloved mother, whose presence has made our lives meaningful. I also thank my siblings, Melike and Gökçe, and my grandmother Gülhanım ORHAN, who has always been our supporter as the eldest in our family.

I am also thankful to TUBITAK for providing financial support through the project 1649B022204720 under the “2210-C National MSc Scholarship Program”.

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1. MicroRNA (miRNA) 2
1.2. Transfer RNA (tRNA) 3
1.3. Circular RNA (circRNA) 3
1.4. Long Non-Coding RNA (lncRNA) 4
1.5. Machine Learning Algorithms in non-coding RNA identification 4
1.6. Aim of Study 5

2. METHOD 8

2.1. Negative RNA Generation Framework 8
2.2. Data Collection 11
2.3. Secondary Structures Calculation of RNA Sequences 13
2.4. Mathematical Representation of RNA Sequences 14
2.5. Negative Data Generation Process 15
2.6. Case Studies 17
2.6.1. Feature Creation to Machine Learning Algorithms 17
2.6.2. Generation of Case Studies 18

3. RESULTS AND DISCUSSION 21

3.1. Results 21
3.2. Discussion 27

4. CONCLUSIONS AND FUTURE PROSPECTS 29

4.1. Conclusions 29
4.2. Societal Impact and Contribution to Global Sustainability 30
4.3. Future Prospects 31

5. APPENDIX 40

6. CURRICULUM VITAE 43

LIST OF FIGURES

Figure 1.1.1 Classification of noncoding RNAs (ncRNAs).. 2
Figure 2.1.1 The main structure of the NeRNA framework consists of several nodes 8
Figure 2.1.2 NeRNA generation node and its key components..... 10
Figure 2.1.3 Thesis Main Workflow..... 11
Figure 2.5.1 Example Negative Data Generation with NeRNA..... 17
Figure 3.1.1 Comparative Performance of Machine Learning Classifiers 22
Figure 3.1.2 Optimized CNN and FNN Results. 23
Figure 3.1.3 ROC Curve and AUC Value Graphs..... 24
Figure 3.1.4 Box Plots of Classifier Accuracy Scores..... 25
Figure 3.1.5 Species-Specific Performance Comparison of NeRNA..... 26

LIST OF TABLES

Table 2.2.1 Data Collection of NeRNA 12
Table 2.4.1 Mapping Base Elements to Octal Representation in a Sequence. 14
Table 2.6.1 An explanation of three maps. 18
Table 2.6.2 Components of a 36-Dimensional Vector 18
Table 3.1.1 Comparison of NeRNA Performance with Similar Methods 26
Table 3.1.2 Top 5 Feature Selection Based on Information Gain Ratio. 27

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
AUC	Area Under the Curve
circRNA	Circular RNA
CNN	Convolutional Neural Network
DNA	Deoxyribonucleic Acid
DT	Decision Tree
FNN	Feedforward Neural Network
KNIME	Konstanz Information Miner
lncRNA	Long non-coding RNA
MCCV	Monte Carlo Cross Validation
miRNA	MicroRNA
ML	Machine Learning
MLP	Multilayer Perceptron
mRNA	Messenger RNA
NB	Naïve Bayes
ncRNA	Non-coding RNA
NeRNA	Negative RNA
RF	Random Forest
RNA	Ribonucleic Acid
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
tRNA	Transfer RNA
TSS	Transcription Start Sites

VENI
VIDI
VICI(M)

To my family…

Chapter 1

Introduction

The discovery of non-coding RNAs (ncRNAs) has fundamentally transformed our understanding of gene regulation and expression [1]. Initially, the central dogma of molecular biology posited that RNA served as a temporary intermediary between DNA and proteins, with all genetic information translated into proteins via messenger RNA (mRNA) [2]. This perspective has been fundamentally revised, as emerging evidence reveals that not all RNAs contribute to protein synthesis. In fact, only a small fraction of genes in the human genome, approximately 1.5%–2%, encode proteins [3], [4]. This emphasizes the extensive and complex network of ncRNAs that have essential functions in the cell, going beyond typical genetic processes [5]. Although they do not code for proteins, these non-coding RNAs are crucial in controlling gene expression, promoting cellular differentiation, and influencing the progression of diseases such as cancer and neurological disorders [6], [7], [8], [9], [10].

Non-coding RNAs, which constitute 98% of the human genome, were categorized according to their length. Although there is no clear borderline between the classes, ncRNAs are generally classified as short, small, or long ncRNAs, as illustrated in Figure 1.1.1 [11]. Among short non-coding RNAs, microRNAs (miRNA) are the most well-known, followed by transfer RNAs (tRNAs), which play crucial roles within the cell. In the category of long non-coding RNAs (lncRNA) was identified as the most significant subgroup. Circular RNAs (circRNAs), known for their unique structure, belong to the long non-coding RNA category [12].

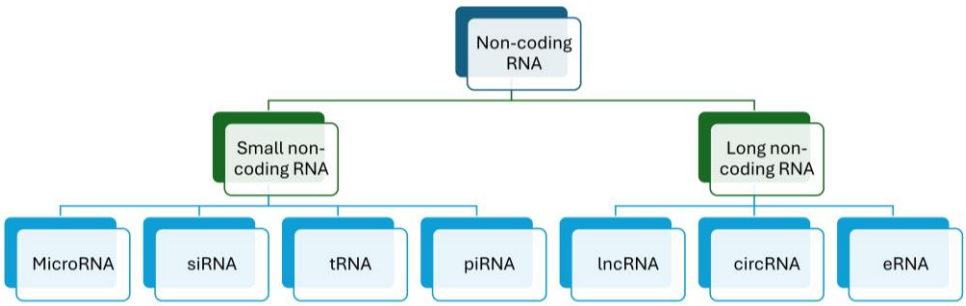

Figure 1.1.1 Classification of noncoding RNAs (ncRNAs). Noncoding RNAs are classified into small ncRNAs (< 200 nucleotides) or long ncRNAs (> 200 nucleotides) based on their length. siRNA, piRNA, eRNA stand for Small interfering RNA, Piwi-interacting RNA, Enhancer RNA respectively.

The intricate network of interactions between coding and noncoding RNAs underscores the complexity of gene regulation in living organisms [13]. It has been demonstrated that these non-coding RNAs are known to regulate gene expression at multiple levels, including transcription, translation, and post-transcriptional modification [5]. Understanding their functions and interactions is essential to unraveling the complexities of cellular processes and identifying potential therapeutic targets. For instance, miRNAs, a class of non-coding RNAs, have received considerable attention for their regulatory roles in gene expression and their links to various diseases, including cancer, neurological disorders, and developmental anomalies [14]. Furthermore, the ongoing identification of novel ncRNAs is continuously enhancing our understanding of the various mechanisms involved in gene regulation.

1.1. MicroRNA (miRNA)

MicroRNAs (miRNAs), one of the studied classes of short non-coding RNAs about 18–22 nucleotides in length, play an essential role in the regulation of gene expression at the post-transcriptional level [15], [16], [17]. By binding to target mRNA transcripts, miRNAs modulate gene expression through translational repression or mRNA degradation. Thousands of miRNAs have been identified across species, underscoring their significance as critical regulators of gene expression. They are involved in the normal functioning of eukaryotic cells and a variety of processes, such as cell differentiation, proliferation, growth, and apoptosis [18], [19]. In addition, they have been implicated in various diseases, including ocular diseases, cardiovascular diseases,

immune disorders, neurodegenerative diseases, leukemia, and epilepsy. Moreover, miRNAs have been identified as crucial players in cancer, influencing tumor cell states, resistance, angiogenesis, and progression [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30]. The creation of miRNA mimics and inhibitors shows how miRNAs can be used to treat diseases and opens up new treatment options that may be better than or in addition to current methods [31].

1.2. Transfer RNA (tRNA)

Transfer RNA (tRNA) is a class of non-coding RNA molecules that are vital for protein synthesis. They act as adaptors, translating the genetic code of mRNA into amino acids, which are the building blocks of proteins [32], [33]. Approximately 600–700 tRNA genes have been identified in the human genome [34]. These molecules ensure accurate and efficient protein synthesis by matching the mRNA codons with their corresponding amino acids. Their unique structure, called cloverleaf, has helped characterize its structure [35]. tRNAs are the modified RNA molecules, with a diverse range of post-transcriptional modifications that are crucial for their translation function [36], [37]. In addition to their well-known role in translation, tRNAs also play a significant role in regulating various cellular processes such as stress responses, cell proliferation, and genome stability [38]. Because of their essential roles in different cellular processes, tRNAs have become potential biomarkers and therapeutic targets in various diseases [39], [40]. Ongoing research continues to uncover the complex roles of tRNA in cellular function and disease.

1.3. Circular RNA (circRNA)

Circular RNAs (circRNAs) are a unique class of non-coding RNA characterized by their covalently closed loop structures [41]. CircRNAs play complex roles in regulatory networks and affect gene expression and disease mechanisms [42]. They can act as miRNA sponges, influence gene expression through interactions with RNA-binding proteins, and potentially regulate protein translation [43]. In addition, circRNAs have been linked to many diseases, such as tumors, neurological disorders, diabetes, vascular diseases, and plant stress responses [44], [45], [46]. Their stability and specific expression patterns make them promising candidates for use as biomarkers and therapeutic targets.

1.4. Long Non-Coding RNA (lncRNA)

Long noncoding RNAs (lncRNAs) are an essential class of non-coding RNA molecules longer than 300 nucleotides [47]. These molecules are involved in various biological processes, including regulation of gene expression at the epigenetic, transcriptional, and post-transcriptional levels, as well as playing roles in chromatin remodeling, genomic imprinting, and cell cycle regulation. lncRNAs play crucial roles in physiological and pathological conditions, including cancer progression and development [47], [48], [49]. Aberrant lncRNA expression has been observed in thyroid, lung, and hematological malignancies and laryngeal and other neoplasms, suggesting their potential as biomarkers for diagnosis, prognosis, and therapeutic targets. The functional diversity and abundance of lncRNAs highlight the need for active investigation of their regulatory mechanisms and promise of novel diagnostic and therapeutic strategies [50], [51].

Rapid development of sequencing technologies has generated vast amounts of RNA data, necessitating practical computational approaches for ncRNA identification, classification, and functional prediction [52]. Machine learning has emerged as a powerful tool for mining complex biological data, offering the potential to predict novel ncRNAs that may play a role in gene regulation or essential functions in organisms [53].

1.5. Machine Learning Algorithms in Non-Coding RNA Identification

Machine learning (ML) is based on the idea that an algorithm can mimic human learning processes and extract rules to generate models [54]. ML has become a popular method in bioinformatics applications as there are many biological fields, such as genome, systems biology, evolution, microarray, and proteomics, where it can extract information from data [55]. ML has gained prominence in bioinformatics, a field rich in applications ranging from genomics and systems biology to evolution, microarrays, and proteomics, where it extracts meaningful information from complex datasets [55]. As a suite of algorithms, ML elucidates the underlying characteristics of any dataset, aiding humans or machines in the decision-making process [54]. For instance, researchers at the University of Nottingham have applied ML algorithms for cardiovascular disease risk assessment, outperforming experienced medical doctors in predicting heart attack risk [56]. Further applications of ML span various domains, including identifying RNA-

binding sites, intracellular localization of proteins, detection of transcription start sites (TSSs) that are pivotal in disease genesis, and discerning catalytic residues and protease cleavage sites from sequence data [57].

Given the significant role of ncRNAs in cellular processes and the experimental challenges in their comprehensive identification, developing reliable computational strategies for ncRNA detection is crucial. Computational prediction, a cost-effective and less labor-intensive alternative to experimental methods, has become integral to ncRNA research [57]. Machine learning-based algorithms have been widely utilized in various ways. Although unsupervised approaches have been employed in some instances, most machine learning applications for ncRNA prediction rely on supervised learning techniques that involve training a classification algorithm using known examples [58]. These approaches typically involve collecting datasets of RNA sequences and structures to compute the features that define the characteristics of the targeted RNA class. Subsequently, a classifier was trained to formulate rules based on the input data, including examples of positive (non-coding RNAs) and negative (coding RNAs). The model produced by the classifier was then applied to the unknown samples and classified accordingly [58].

In bioinformatics, supervised ML algorithms are employed to learn from labeled biological data, enabling the prediction or classification of new unseen data. These algorithms operate on datasets with input-output pairs, using known outputs to train the model. Once trained, the model can predict the output of new inputs based on learned patterns [59]. Although supervised ML is widely used in bioinformatics for various tasks, such as using support vector machines (SVM) for cancer identification through gene expression analysis, its efficacy can vary across applications [60], [61].

1.6. Aim of Study

In a 2013 study [62], it was observed that the class imbalance problem in learning and test data had a negative effect on the prediction of microRNAs. This class imbalance problem has become a problem not only for miRNA prediction applications but also for all applications using binary classification algorithms [63]. In these binary classification applications, researchers perform model setups based on comparisons, such as presence/absence, yes/no, and positive/negative. Therefore, the distribution of positive and negative data directly affects the accuracy of the models. Because negative data are

shared very little in research on biological sequences, it is seen that the class imbalance problem causes more significant problems in machine learning models used in bioinformatics [64].

The success of an ML system relies heavily on obtaining high-quality datasets, feature selection, algorithm selection, training scheme establishment, and parameter tuning [65], [66]. Experimentally validated RNA sequences can serve as a positive class for the classification of ncRNAs. However, the generation of high-quality negative examples lacks standardized methodologies. To solve this problem, researchers have attempted to obtain new negative sequences by randomly changing the indices within the DNA sequences or using sequences belonging to a different class than the type of sequence used in the study [66], [67]. However, this leads to reduced accuracy and overfitting problems in machine-learning models. To address this issue, mathematical and thermodynamic principles are based on the novel negative data generation workflow developed in this thesis.

Chapter 2 introduces the development of a novel framework for negative data generation utilizing the Konstanz Information Miner (KNIME) workflow management system [68] and RNAfold software [69] for secondary structure calculations. The methodology began with the collection of four non-coding RNA types to test and develop NeRNA algorithms. This emphasizes the importance of mathematical representation and secondary structures in generating negative data. By applying a novel shifting process and converting mathematical representations back into sequences, the NeRNA algorithm successfully generated negative data. This process was then evaluated using six different machine learning classifiers, along with a novel set of 36-dimensional features specific to non-coding RNA, with the aim of enhancing model accuracy.

Chapter 3.1 presents the results of the novel NeRNA methodology. It highlights the accurate prediction of non-coding RNA data and the significant improvement in machine learning algorithm performance when utilizing NeRNA-derived negative data compared to literature data. Novel features and an equalized environment contribute to this enhancement. Additionally, a species-based analysis tested the NeRNA framework, demonstrating its efficacy in achieving high accuracy in two-class classification tasks.

Chapter 3.2 critically evaluates and discusses the findings gathered throughout this thesis. This section identifies the limitations of the proposed algorithm and explores the underlying causes of these constraints. Chapter 4 delves into the challenges faced by

the NeRNA method, proposing potential improvements and applications for future research.

Chapter 2

Method

2.1. Negative RNA Generation Framework

The negative RNA generation framework (NeRNA) was developed within the Konstanz Information Miner (KNIME) workflow management system [68]. The framework consists of several key components (Figure 2.1.1), in which the initial step involves the input of a sequence via a sequence file reader node (Figure 2.1.1). This node is accepting the sequences in either the FASTA or fa file format, and users must be specifying the type of RNA sequence that will be processed by the framework. Additionally, the PATH of the RNAfold software [69], which is essential for secondary structure calculations, must be configured (PATH of the RNAfold varying from the operating system to the system). In addition to user configuration, the node has a default configuration, including the miRNA hairpin FASTA file (hairpin.fa), with the sequence type set to miRNA, and the PATH of the RNAfold set for the Linux operating system.

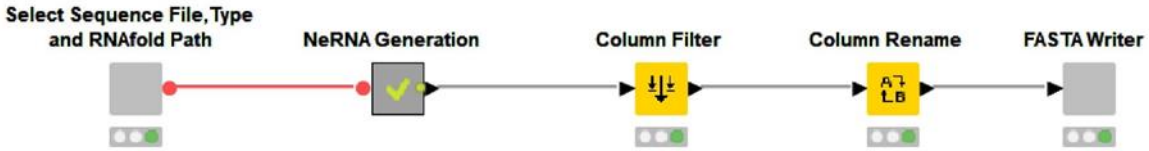

Figure 2.1.1 The main structure of the NeRNA framework consists of several nodes. The "Select Sequence File, Type, and RNAfold Path" node serves as an entry point. This node is followed by the main node called "NeRNA Generation." The exit node of this framework is a FASTA writer, which converts negative sequences into the FASTA format.

The second and most significant component of the NeRNA framework is referred to as NeRNA generation. It comprises two subgroups: Secondary Structure

Calculation and NeRNA, as shown in Figure 2.1.1. The selection of the sequence type, which is taken from the user, determines the case switch part effect on the secondary structure calculation and the NeRNA algorithm parameters. For instance, if the user selects the circRNA option, the --circ parameters will be applied to the secondary structure calculation process. Instead, if the user selects the tRNA sequence type, the NeRNA node will be modified accordingly.

RNAfold software was used to calculate secondary structures in the NeRNA Generation component (Figure 2.1.2). Secondary structures are essential for the generation of negative RNA sequences (Detailed in Section 2.3). This component includes R scripts that take the sequence file and RNAfold PATH, which are provided by the user. Additionally, this part of the R scripts incorporates the seqinR (v4.2.8) [70] and stringr (v1.4.0) packages, which play crucial roles in reading and manipulating biological sequences.

The third component is the main node of our framework is called NeRNA algorithm. It possesses two essential functions: the sequence converter (Figure 2.1.2-a) and the negative generator binary index change (Figure 2.1.2-b). The sequence converter transforms biological sequences into mathematical representations, while the negative generator binary index change generates novel negative sequences from the original sequence and its secondary structure (Details will be introduced in Section 2.5). The final component of our framework is the FASTA writer (Figure 2.1.1), which generates a FASTA file containing NeRNA-derived negative sequences. This component is configurable based on user preferences, including the output location, file name, and sequence extension (fa or FASTA).

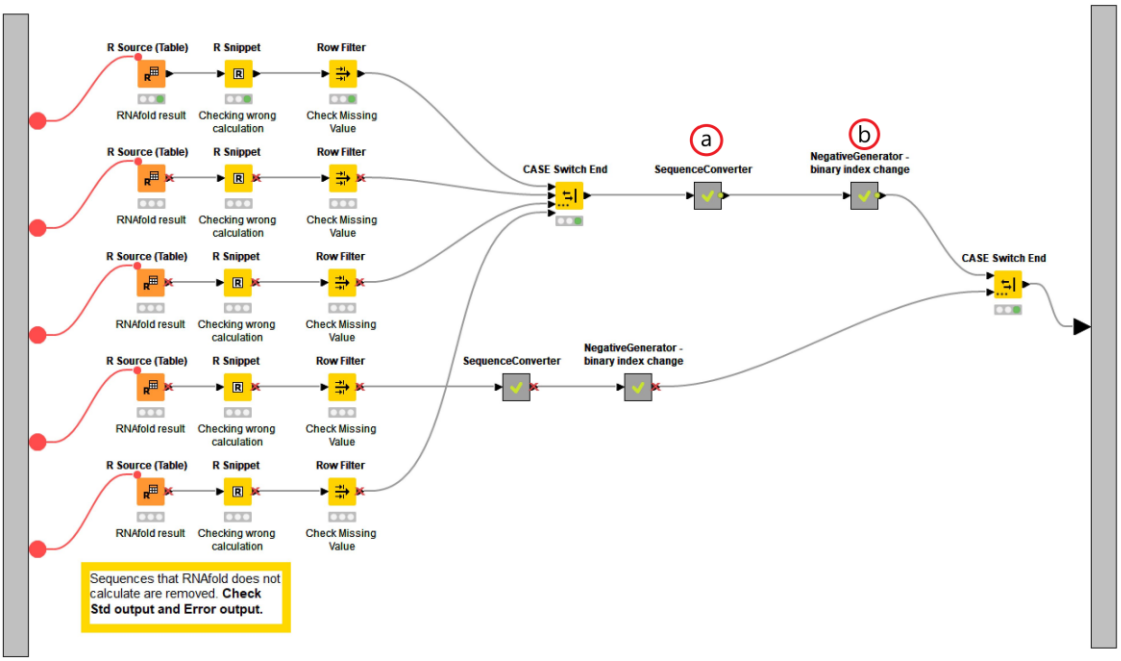

Figure 2.1.2 NeRNA generation node and its key components. R Source, compute RNA secondary structures using R programming. a- Sequence Converter Node, Converts biological sequences into a mathematical representation. b- Negative Generator, primary node responsible for generating negative RNA sequences.

In addition to our primary algorithms, a case studies' workflow was established with the purpose of evaluating the performance of the NeRNA algorithm. Additionally, the case studies' workflow was improved with six different machine learning and deep learning algorithms. This workflow was created using the default algorithms of KNIME and external libraries including, Keras [71] and TensorFlow [72] (for further information, please refer to the Section 2.6). Moreover, custom R and Python scripts were developed to perform specific tasks utilizing the seqinR, stringr, Keras, and TensorFlow packages.

The process of generating negative data relies on an RNA sequence, as well as secondary structures of positive data and chemical properties such as minimum free energy. The process of negative data generation involves four steps (Figure 3):

1- Secondary Structures Calculation of RNA Sequences
2- Mathematical representation of RNA sequences
3- Negative data generation process
4- Case Studies

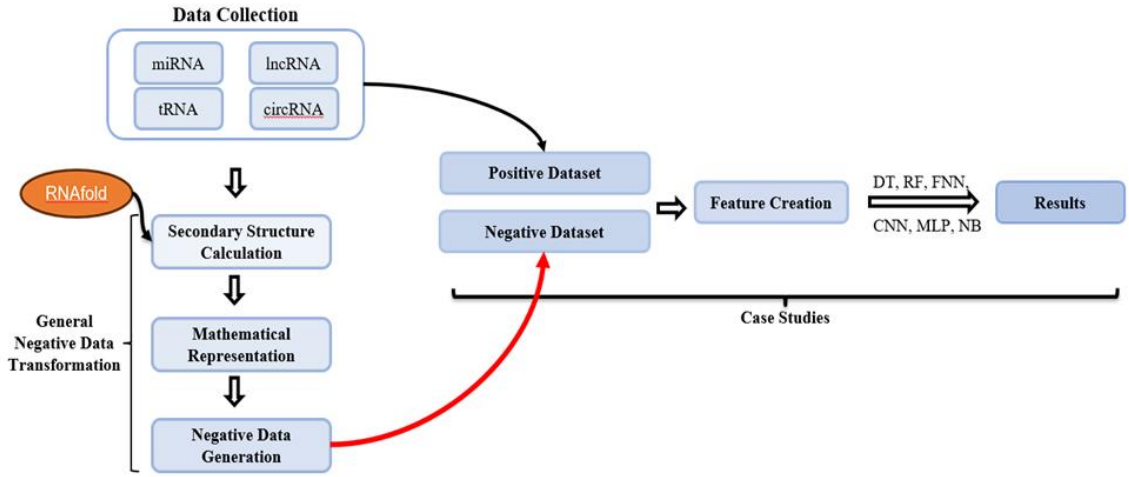

Figure 2.1.3 Thesis Main Workflow. Data Collection, General Transformation, Case Studies with Decision Tree (DT), Random Forest (RF), Feedforward Neural Network (FNN), Convolutional Neural Network (CNN), Multilayer Perceptron (MLP), Naïve Bayes (NB), and RNAfold for secondary structure calculations.

2.2. Data Collection

Four distinct groups of ncRNA sequences were compiled from multiple databases to perform testing and develop novel approach the NeRNA algorithm. The initial dataset comprised 9315 miRNA hairpin sequences obtained from several species including *Mus musculus, Homo sapiens, Gallus gallus, Monodelphis domestica, Oreochromis niloticus, Glycine max, Equus caballus, Bos taurus, Medicago truncatula,* and *Pan troglodytes* from the miRBase database (v22.1) [73], with an average sequence length of 85.57.

The second group comprised 1000 randomly selected human lncRNAs from LNCipedia [74], with an average length of 1496.97. The third group included 1000 randomly selected mouse circular RNA sequences from circBase [75], with an average length of 1566.49. Lastly, the fourth group consisted of 1110 tRNA sequences derived from 101 distinct species with an average length of 77.56. These sequences, together with their pre-calculated secondary structures obtained from the T-Psi-C database [76], were included in the analysis. (Table 1).

Testing and development of NeRNA algorithms have been synchronously studied. We used human miRNA sequences to develop the main functionality of the workflow. Then, well-prepared workflow was extended to four distinct data groups for the improvement of the algorithm. The primary purpose of selecting these data groups

was that they are well-studied ncRNA groups (tRNA, lncRNA, circRNA, and miRNA), which possess unique secondary structure shapes such as cloverleaf, linear, circular, and hairpin-like structures, respectively [77]. Furthermore, their overall length is distinct from each other, in our case, lncRNA and circRNA average length are longer than 1400 nucleotides, while miRNA and tRNA average lengths were shorter than 100 nucleotides. This diverse data collection allows us to compare the efficiency of our algorithm using various sizes and different structures, as well as an examination of the effect of sequences from different sources on the algorithm.

Table 2.2.1 Data Collection of NeRNA. The number indicates the count of sequences in each dataset. The list of 101 different organisms for the tRNA dataset is available in Appendix.

Datasets	**RNA type**	**Organisms**	**Number**	**Sequence Length**			**Source**
				Min	Max	Average	
I	miRNA hairpins	*Homo sapiens*	1917	41	180	81.89	miRBase[78]
		Mus musculus	1234	39	147	82.60	
		Bos taurus	1064	43	149	76.23	
		Gallus gallus	882	48	169	87.36	
		Oreochromis niloticus	812	40	100	61.05	
		Equus caballus	715	52	145	104.61	
		Glycine max	684	54	473	135.92	
		Monodelphis domestica	680	44	111	64.92	
		Medico truncatula	672	54	910	165.26	
		Pan troglodytes	655	69	148	89.94	
II	lncRNA	*Homo sapiens*	1000	202	29066	1496.97	LNCipedia[79]
III	circRNA	*Mus musculus*	1000	51	29991	1566.49	circBase[80]
IV	tRNA	101*	1110	54	99	77.56	T-Psi-C [81]

2.3. Secondary Structures Calculation of RNA Sequences

RNA sequences were mainly composed of four bases (A, G, C, and U) that can form base pairs, G-C, A-U, and G-U [82]. These pairs play a crucial role in determining the secondary structure of the RNA molecules [83]. The energy required for the formation of each pair varies. To predict the secondary structures of RNA molecules, various algorithms have been developed based on the principle of minimum free energy [84], [85], [86], [87], [88], [89]. In this study, RNAfold software from the ViennaRNA package was employed to generate secondary structures [69]. The RNAfold was embedded in the second component of the framework (Figure 2.1.1). The node takes the PATH of RNAfold and the sequence file from the users and implements this information in R scripts that calculate the secondary structures. After calculation, the output data table contains the sequence, dot-bracket representation of secondary structures [90], [91], and minimum free energy. Additionally, the RNAfold software has several parameters for generating structures, such as --noGU, --noLP, and --temp, which respectively mean "Do not allow GU pairs", "Produce structures without lonely pairs", and "Rescale energy parameters to a temperature of temp °C. Default is 37 °C". In our study, RNAfold was utilized with the "Do not produce postscript drawing of the mfe structure" (--noPS) parameters for miRNA, lncRNA, tRNA, and circRNA datasets, and the circular structures generation option (--circ) was also applied for the circRNA dataset which contains circRNA sequences.

Before mathematical representation, our framework will filter sequences that do not calculate minimum free energy and are more than 30,000 nucleotides in length, due to the RNAfold limitations. This process is embedded within an R script that takes into account the PATH and sequences, which will be sent sequence by sequence for the creation of a calculation table. The script works in two essential libraries: seqinR and stringr. The Read.fasta function reads the sequences and stores them in an R-list object. The run_RNAfold function sends the R list to RNAfold with a parallel calculation to enhance the time efficiency. The resulting data table contains crucial information, including secondary structures and the minimum free energy of the RNA sequences.

To understand the secondary structures of RNA molecules in a better way, RNAfold uses a dot-bracket representation in which nucleotides involved in base pairs are shown as brackets, while non-paired nucleotides are shown as dots [90], [91]. In our

algorithm, we converted this representation to show base pairs as A, G, C, and U and non-base pairs as A ', G ', C ', and U '. This conversion is conducted in the sequence converter component of the NeRNA algorithm (Figure 2-2). The calculated structure table, which includes the RNAfold results, is processed sequence by sequence in this component. The table is first split into individual nucleotides and then transposed using the dot-bracket representation. The first column of the transposed table covers the bases, while the second column provides pairing information for each base. A one-to-one match was created to combine the pair information and the bases, allowing for a more detailed understanding of secondary structures, and facilitating the conversion of these structures into mathematical representations.

2.4. Mathematical Representation of RNA Sequences

The architecture of the NeRNA algorithm was developed based on mathematical representations of the biological principles of RNA folding. This mathematical representation of biological sequences is an important approach in the application of machine learning. Without this, algorithms cannot learn directly from biological letters, leading to a decrease in the precision of the classification algorithms. Therefore, we developed a binary-based octal representation of RNA sequences (Table 2).

Table 2.4.1 Mapping Base Elements to Octal Representation in a Sequence.

Base	Octal Representation
A	000
G	001
C	010
U	100
A'	011
G'	110
C'	101
U'	111

In our study, we utilized RNA sequences that comprise secondary structure information in a data table. Subsequently, each sequence was separated into nucleotides and converted into its corresponding three-digit numerical representation based on our

predefined map (Table 2.4.1). Following this, the table was re-transformed and combined into a single, numerical format.

The octal representations are taken from the secondary structures of sequences upon conversion of the base elements to their octal representations using our conversation map. (See in detail section 2.5.) (Table 2.4.1)

2.5. Negative Data Generation Process

An overview of the algorithm the generation of negative sequences is shown in Algorithm 2.5.1. The RNA sequence, with length n, was obtained using this algorithm (Figure 2.5.1). Afterwards, the secondary structure of the sequence was determined using the RNAfold package in ViennaRNA. The calculated secondary structures were subsequently transformed into A', G,' C', and U' or A, G, C, and U, based on to their dot-bracket notation.

At the beginning of the Negative RNA generation process, each base element of the sequence was transformed into numerical data by using a predefined base-to-octal element mapping method (Table 2.4.1). In our mapping, each base element is represented by a three-digit number, hence the length of the numerical representation is 3n.

The process involves creating an octal representation of a new sequence, where the first element is moved to the last index and becomes the 3nth entry in the newly created representation (Figure 2.5.1 (A)). The indices for all the other entries are then decreased by one, resulting in a shifted representation. Thus, all the base proportions of the sequences change.

This frameshift-like shifting process creates a novel numerical value that can be transformed inversely using the octal-to-base transformation (Table 2.4.1). The outcome of this inverse transformation process is a secondary structure-forming sequence that is not necessary. Therefore, the sequence was converted to an RNA sequence with no specific structural information. The final sequence obtained was a novel negative RNA sequence (Figure 2.5.1). This process was repeated for each sequence in the NeRNA node of our framework.

Algorithm 1: Generate Negative Sequences

Input: RNA Sequence file R

Output: Negative Sequences Table

for *each* $x \in R$ **do**

  ▷ x: one sequence

  ▷ n_x: length of x

  ▷ Step 1: RNAfold Calculation for x

  f_{fold} = secondary structure of x

  for *each* $i \in \{1, 2, 3, \ldots, n_x\}$ **do**

    ▷ i: index of base

    if $f_{fold_i} =$. **then**

      $x_i = x_i'$

    if $f_{fold_i} =$ (**or**) **then**

      $x_i = x_i$

  ▷ Step 2: Octalization Method

  for *each* $i \in \{1, 2, 3, \ldots, n_x\}$ **do**

    ▷ y: octal representation of x

    $(y_{3i-2}y_{3i-1}y_{3i}) = T(x_i)$ ▷ T:(table)

  ▷ Step 3: Shifting on y

  ▷ z: shifted version of y

  $z_{3n_x} = y_1$

  for *each* $j \in \{1, 2, 3, \ldots, 3n_x - 1\}$ **do**

    $z_j = y_{j+1}$

  ▷ Step 4: Inverse of Octal Representation

  ▷ w: inverse octal representation of z

  for *each* $i \in \{1, 2, 3, \ldots, n_x\}$ **do**

    $w_i = T^{-1}(z_{3i-2}z_{3i-1}z_{3i})$

    if $w_i = A'$ **then**

      $w_i = A$

    if $w_i = G'$ **then**

      $w_i = G$

    if $w_i = U'$ **then**

      $w_i = U$

    if $w_i = C'$ **then**

      $w_i = C$

  Concatenate w_i

Output negative sequences table

Algorithm 2.5.1 Algorithm of NeRNA.

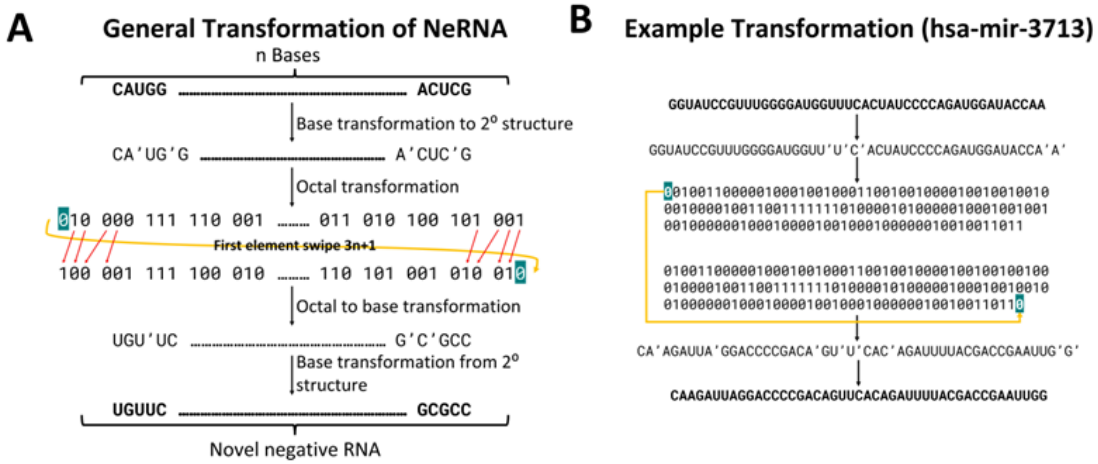

Figure 2.5.1 Example Negative Data Generation with NeRNA - (A) General transformation of NeRNA with any sequence, (B) Transformation of hsa-mir-3713.

2.6. Case Studies

Four case studies were conducted to test our approach with non-coding RNAs. The primary objective of these case studies was to show how NeRNA enhances the generation of negative examples and demonstrates the impact of these negative sequences on the accuracy of classification model predictions.

For this purpose, case studies were developed to evaluate the efficiency of the workflow, which includes positive sequence, negative sequence generation, combined data collection, cross-validation, and machine learning model testing. (Figure 2.1.3). In addition, we have incorporated a unique feature for the machine learning process in these case studies, which is a 36-dimensional vector feature that we have developed. This feature has contributed to the precision of the machine learning models.

2.6.1. Feature Creation to Machine Learning Algorithms

The creation of graphical features based on RNA secondary structures was accomplished by Zhang et al. Their 3D representation was based on the chemical properties of each base in the RNA structure [92]. The bases were grouped based on their chemical properties, with purine group R consisting of A and G, and pyrimidine group Y consisting of C and U. Additionally, there were weak H-bond group W consisting of A and U, and strong H-bond group S consisting of C and G. The amino group M is consisted of A and C, and the keto group K is consisted of G and U. According to the base classification scheme (i), (ii), and (iii), a characteristic sequence can be depicted through

three maps, $a_1, a_2,$ and a_3 respectively (Table 1). The sequence is represented by its 3D graph in the characteristic sequence from the first base to the *i-th* base, where *i* ranges from 1 to *n*, and *n* represents the length of the characteristic sequence.

The same approach to base grouping was applied to the map we defined, which contained alpha1, alpha2, and alpha3 groups, where n is the length of the RNA sequence, and i is the index of the base in the sequence (Table 2.6.1). A unique approach was used for each primer number in our maps. However, due to the number limitation of the KNIME platform, prime numbers 101, 103, 107, and 109 were optimized on a 1/100 scale. These individualized maps contribute to a unique representation of each sequence. Expanding our focus to non-coding RNAs, we generated 3 unique maps, resulting in a 36-dimensional vector feature set, as detailed in Table 2.6.2. This 36-dimensional feature set was calculated for both the positive and negative classes in the case studies without employing any normalization or scaling methodology.

Table 2.6.1 An explanation of three maps.

$a_1(g_i) = (x_{1i}, y_{1i}, z_{1i})$				$a_2(g_i) = (x_{2i}, y_{2i}, z_{2i})$				$a_3(g_i) = (x_{3i}, y_{3i}, z_{3i})$			
g_i	x_{1i}	y_{1i}	z_{1i}	g_i	x_{2i}	y_{2i}	z_{2i}	g_i	x_{3i}	y_{3i}	z_{3i}
$\{A\ or\ C\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.01^i	$\{A\ or\ G\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.03^i	$\{A\ or\ U\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.07^i
$\{G\ or\ U\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.03^i	$\{C\ or\ U\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.07^i	$\{C\ or\ G\}$	$\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.09^i
$\{A'\ or\ C'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.07^i	$\{A'\ or\ G'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.09^i	$\{A'\ or\ U'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$\cos\left(\frac{2\pi i}{n}\right)$	1.01^i
$\{G'\ or\ U'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.09^i	$\{C'\ or\ U'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.01^i	$\{C'\ or\ G'\}$	$-\sin\left(\frac{2\pi i}{n}\right)$	$-\cos\left(\frac{2\pi i}{n}\right)$	1.03^i

Table 2.6.2 Components of a 36-Dimensional Vector

$$x_1^1=\frac{1}{n}\sum_{i=1}^{n}x_{1i}^{A,C},\quad y_1^1=\frac{1}{n}\sum_{i=1}^{n}y_{1i}^{A,C},\quad z_1^1=\frac{1}{1.01^n}\sum_{i=1}^{n}z_{1i}^{A,C},\quad x_1^2=\frac{1}{n}\sum_{i=1}^{n}x_{1i}^{G,U},\quad y_1^2=\frac{1}{n}\sum_{i=1}^{n}y_{1i}^{G,U},\quad z_1^2=\frac{1}{1.03^n}\sum_{i=1}^{n}z_{1i}^{G,U},$$

$$x_1^3=\frac{1}{n}\sum_{i=1}^{n}x_{1i}^{A',C'},\quad y_1^3=\frac{1}{n}\sum_{i=1}^{n}y_{1i}^{A',C'},\quad z_1^3=\frac{1}{1.07^n}\sum_{i=1}^{n}z_{1i}^{A',C'},\quad x_1^4=\frac{1}{n}\sum_{i=1}^{n}x_{1i}^{G',U'},\quad y_1^4=\frac{1}{n}\sum_{i=1}^{n}y_{1i}^{G',U'},\quad z_1^4=\frac{1}{1.09^n}\sum_{i=1}^{n}z_{1i}^{G',U'},$$

$$x_2^1=\frac{1}{n}\sum_{i=1}^{n}x_{2i}^{A,G},\quad y_2^1=\frac{1}{n}\sum_{i=1}^{n}y_{2i}^{A,G},\quad z_2^1=\frac{1}{1.03^n}\sum_{i=1}^{n}z_{2i}^{A,G},\quad x_2^2=\frac{1}{n}\sum_{i=1}^{n}x_{2i}^{C,U},\quad y_2^2=\frac{1}{n}\sum_{i=1}^{n}y_{2i}^{C,U},\quad z_2^2=\frac{1}{1.07^n}\sum_{i=1}^{n}z_{2i}^{C,U},$$

$$x_2^3=\frac{1}{n}\sum_{i=1}^{n}x_{2i}^{A',G'},\quad y_2^3=\frac{1}{n}\sum_{i=1}^{n}y_{2i}^{A',G'},\quad z_2^3=\frac{1}{1.09^n}\sum_{i=1}^{n}z_{2i}^{A',G'},\quad x_2^4=\frac{1}{n}\sum_{i=1}^{n}x_{2i}^{C',U'},\quad y_2^4=\frac{1}{n}\sum_{i=1}^{n}y_{2i}^{C',U'},\quad z_2^4=\frac{1}{1.01^n}\sum_{i=1}^{n}z_{2i}^{C',U'},$$

$$x_3^1=\frac{1}{n}\sum_{i=1}^{n}x_{3i}^{A,U},\quad y_3^1=\frac{1}{n}\sum_{i=1}^{n}y_{3i}^{A,U},\quad z_3^1=\frac{1}{1.07^n}\sum_{i=1}^{n}z_{3i}^{A,U},\quad x_3^2=\frac{1}{n}\sum_{i=1}^{n}x_{3i}^{C,G},\quad y_3^2=\frac{1}{n}\sum_{i=1}^{n}y_{3i}^{C,G},\quad z_3^2=\frac{1}{1.09^n}\sum_{i=1}^{n}z_{3i}^{C,G},$$

$$x_3^3=\frac{1}{n}\sum_{i=1}^{n}x_{3i}^{A',U'},\quad y_3^3=\frac{1}{n}\sum_{i=1}^{n}y_{3i}^{A',U'},\quad z_3^3=\frac{1}{1.01^n}\sum_{i=1}^{n}z_{3i}^{A',U'},\quad x_3^4=\frac{1}{n}\sum_{i=1}^{n}x_{3i}^{C',G'},\quad y_3^4=\frac{1}{n}\sum_{i=1}^{n}y_{3i}^{C',G'},\quad z_3^4=\frac{1}{1.03^n}\sum_{i=1}^{n}z_{3i}^{C',G'}.$$

2.6.2. Generation of Case Studies

In the case studies section, NeRNA workflow is employed to investigate the effectiveness of NeRNA in generating negative samples for biological sequences. The application of the NeRNA workflow to create artificial sequences utilizing sequence samples from lncRNAs, circRNAs, tRNA, and miRNAs was demonstrated. Moreover, a well-established pseudo-human miRNA dataset [93] and NeRNA-derived negative human miRNA samples were employed as benchmarks to evaluate the efficiency of NeRNA-derived negative examples.

Six machine learning and deep learning algorithms were developed for the purpose of testing NeRNA-generated negative sequences. In this process, a range of classifiers, including Decision Tree (DT) [94], Naive Bayes (NB), and Random Forest (RF), as well as deep learning-based approaches, such as feed-forward neural networks (FNN), Convolutional Neural Network (CNN), and Multilayer Perceptron (MLP), were employed to facilitate the implementation of NeRNA [95]. All these learning algorithms were designed to classify four different ncRNA datasets.

The entire process, from model generation to predictions for classifying datasets, was executed in KNIME. KNIME, known for its user-friendly interface and customizable environment, was chosen for its convenience. The platform includes configurable machine learning algorithms and supports the Keras and TensorFlow libraries for model creation. However, it is important to note that our focus was not on optimizing the models; hence, default settings were used for all models. Detailed parameters and settings for each algorithm are provided in Appendix section.

One of the most common and detrimental issues in the field of classification models are overfitting and imbalanced datasets [63], [96]. To avoid these problems, it is crucial to use a one-to-one ratio of positive and negative datasets. To accomplish this, all datasets utilized in our case studies were generated in the NeRNA workflow and then applied to machine-learning algorithms. For instance, dataset II comprised 1000 human lncRNA sequence samples, which served as positive samples in our models. Additionally, for each sequence in this dataset, a corresponding negative sample was generated using the NeRNA algorithm, resulting in a total of 1000 negative examples.

The sampling process is generally considered as an effective method for generating the models. It is common to use a distribution ratio of 70-30 or 80-20 for training and testing, respectively [97]. In our study, we used an equal number of balanced samples from the negative and positive datasets and divided them into 70-30 training and testing groups with a labelled information of each sequence.

The training dataset that includes labelled data and 36 features of each sequence was employed in the evaluation of six classifiers: NB, RF, DT, FNN, CNN, and MLP. Furthermore, test performance scores were recorded for each iteration of the dataset. A total of 1000 Monte Carlo cross-validation (MCCV) iterations were performed on the training and testing parts of the dataset to assess the model performance [98]. Key metrics, including recall, precision, sensitivity, specificity, F-measure, and accuracy scores, were measured [99] and recorded for each iteration of MCCV.

The NeRNA derived case study workflow, which incorporates six distinct machine learning models, ensures that identical datasets are utilized in each iteration, thereby providing a fair evaluation of each classifier. In addition, all frameworks are reproducible using the same configuration.

Chapter 3

Results and Discussion

3.1. Results

The present study focuses on the development of NeRNA, a novel methodology for creating negative sequences. This methodology is based on an ab initio approach and utilizes secondary structures of sequences and machine-learning features to generate negative data examples. In addition to other associated methods, NeRNA is released as a user-friendly KNIME workflow that can be downloaded from the GitHub page. (https://github.com/Mehmeteminorhan/NegativeRNA)

Machine learning algorithms present a powerful function for evaluating the impact of negative RNA sequences on the performance of classification models. In this study, we explored various machine learning and deep learning models, including RF, DT, NB, MLP, CNN and FNN, which were applied to four distinct non-coding RNA groups: miRNA, tRNA, lncRNA, and circRNA (Figure 3.1.1). Each classifier was trained on datasets specific to these RNA groups to gain insights into their individual characteristics.

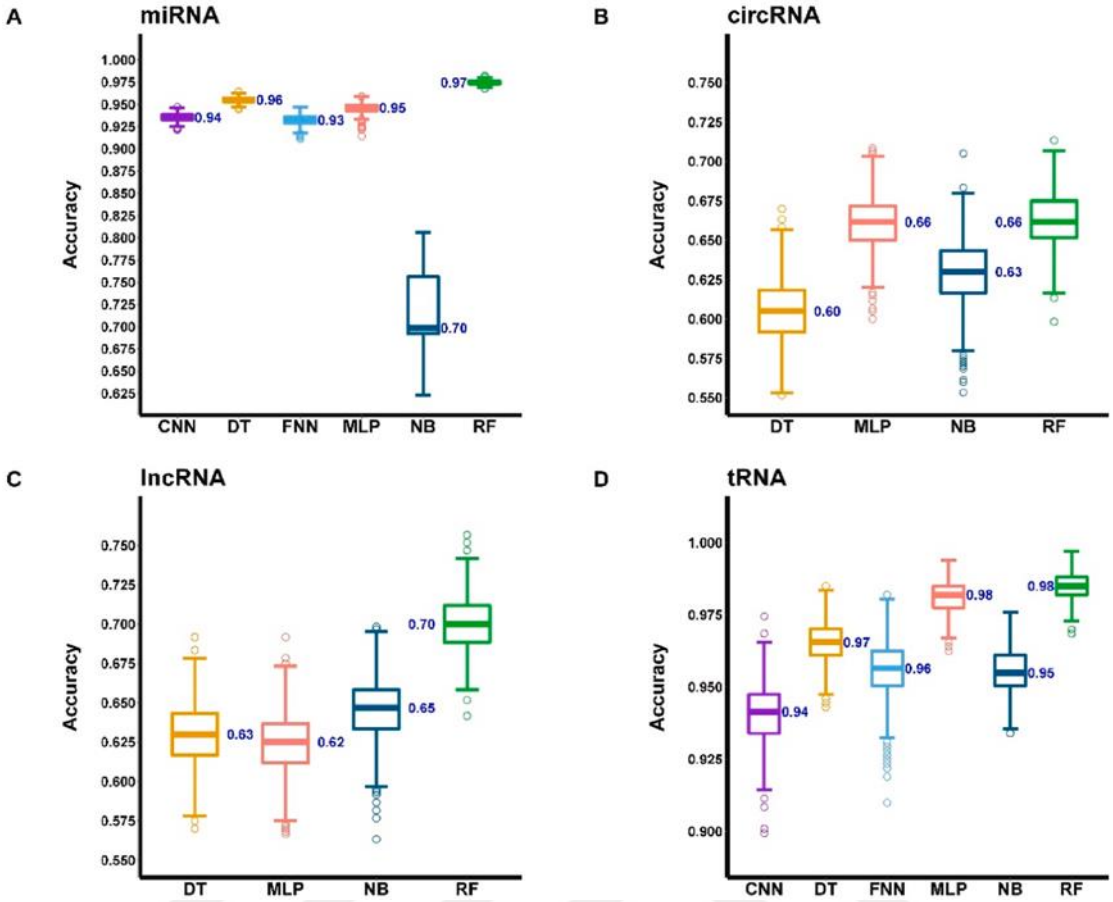

Figure 3.1.1 Comparative Performance of Machine Learning Classifiers. Highlighting Random Forest (RF) as the most effective across miRNA, lncRNA, tRNA, and circRNA categories. The x-axis indicates the classifier, and the y-axis shows accuracy scores. Box-plots represents accuracy values from 1000 models. The values for Q2 (median) percentiles are displayed in the boxes.

Nevertheless, the CNN and FNN models had a problem learning from the data collection of lncRNA and circRNA without normalization or scaling (Figures 3.1.2B, 3.1.2C, and 3.1.2). Consequently, their performance metrics are not presented in Figures Figure 3.1.1B, 3.1.1C, and Figure 3.1.3. So, to get around this problem, decimal scaling normalization [100] was used on the CNN and FNN results on the lncRNA and circRNA datasets. Figure 3.1.2 shows the outcome.

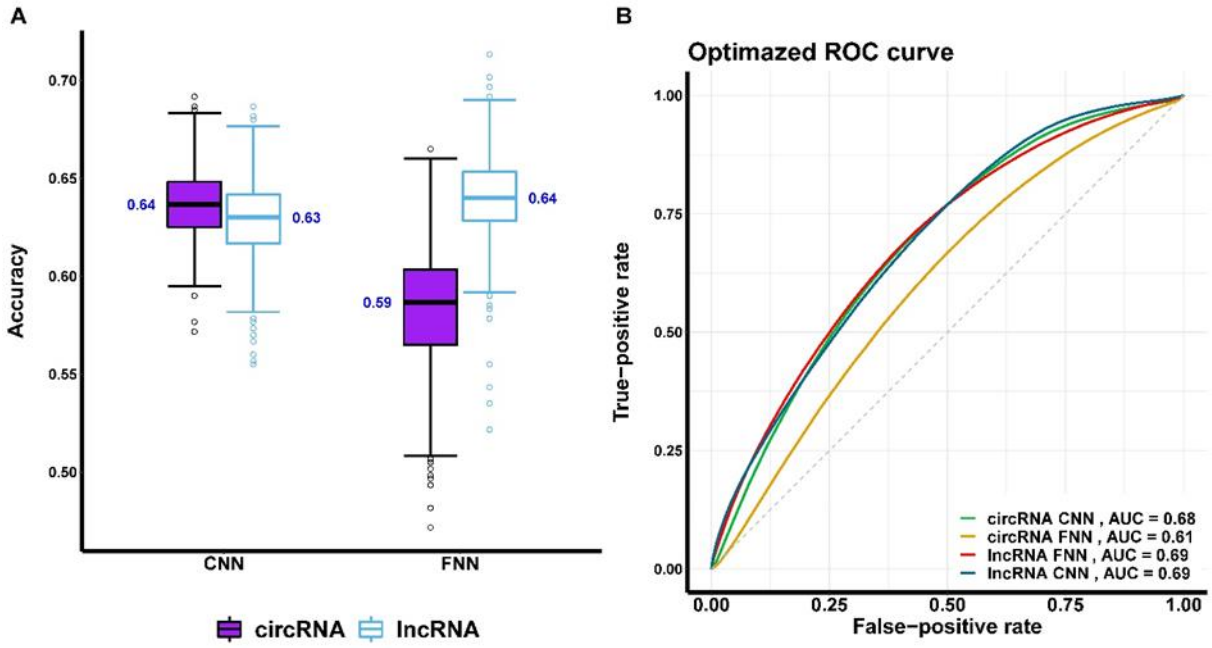

Figure 3.1.2 Optimized CNN and FNN Results for lncRNA and circRNA Datasets, showcasing (A) accuracy values and performance evolution with (B) ROC metrics.

The graphical representation of the statistics of model learning was generated using R libraries, such as ggplot2 [101], ggpubR [102], and pROC [103]. The process involved generating plots for each iteration, where classification metrics and iterations of the MCCV were systematically collected from the KNIME workflow. Subsequently, data wrangling and data visualization were conducted in the RStudio environment.

The case studies demonstrated an overall accuracy of 95 percent for all six classifiers. Among these, the Random Forest algorithm was the most accurate for all non-coding RNA classes. Nevertheless, decision tree models exhibited slightly higher accuracy than multilayer perceptron models in classifying miRNA sequences. However, multilayer perceptron models showed superior performance for circRNA classification.

The models were evaluated using receiver operating characteristic (ROC) curve scores [104]. For this purpose, the probability values of RF, DT, MLP, and NB were collected via KNIME nodes, whereas for the FNN and CNN, probability information was gathered using a two-layer SoftMax function [105]. The ROC curve graphs represent the false positive rate on the x-axis and the true positive rate on the y-axis, and the Area Under the Curve (AUC) represents the accuracy values [104].

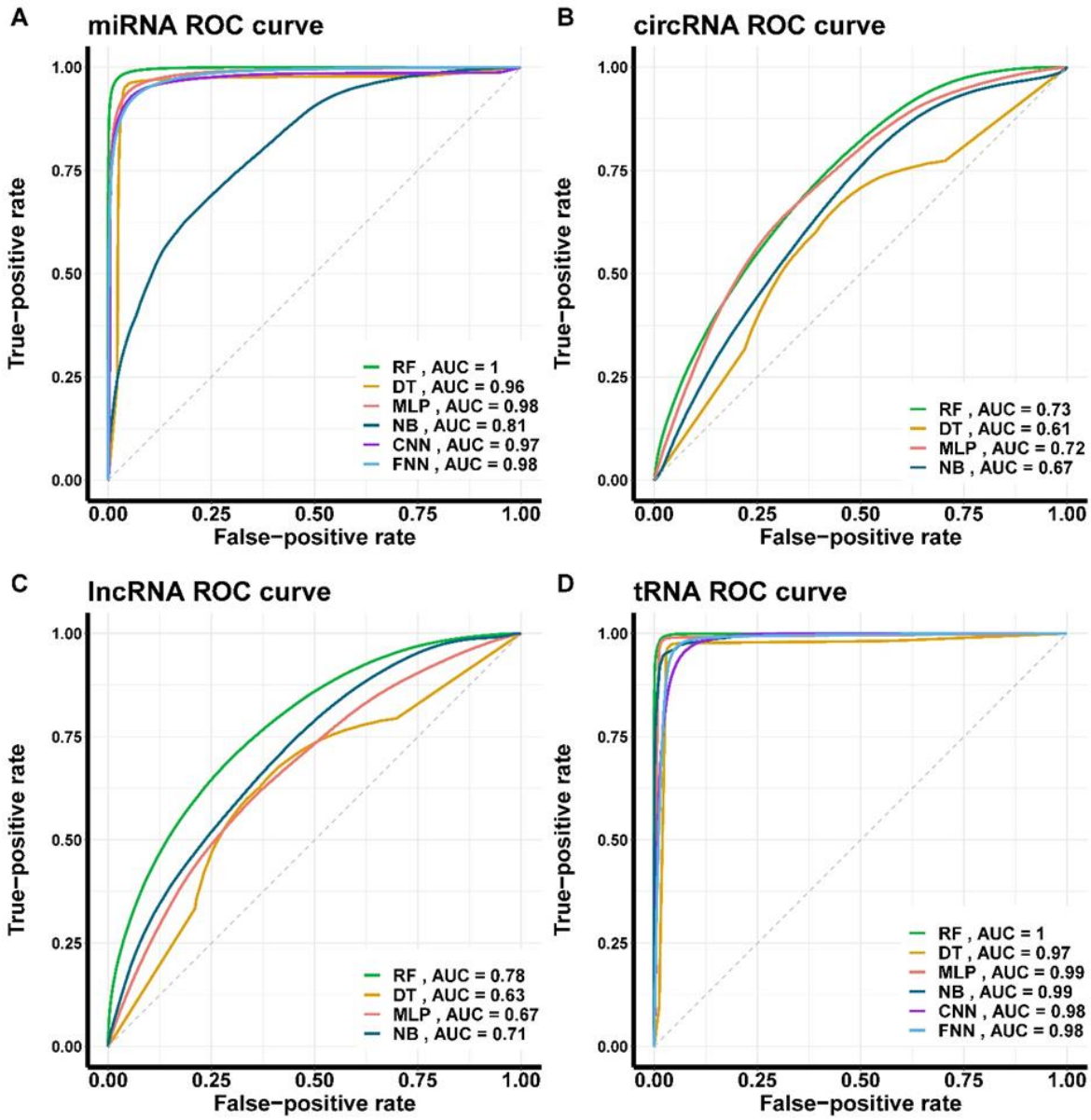

Figure 3.1.3 ROC Curve and AUC Value Graphs. To analysis of CNN, DT, FNN, MLP, NB, and RF Classifiers.

As shown in Figure 3.1.3, Random Forest models achieved higher performance, closely followed by Multilayer Perceptron models. These findings highlight the potential of specific machine learning algorithms trained with NeRNA-derived negative samples to accurately classify classes of non-coding RNAs.

Existing literature lacks well-structured algorithms for generating high-quality negative data. Nevertheless, a few researchers have published their own negative datasets, which they have utilized in their machine-learning models [93], [106]. In order to compare the NeRNA-derived negative sequences with well-known datasets, employing negative human pseudo hairpin sequences was utilized as a benchmark. In this

comparison, the secondary structures and 36-dimensional properties of each sequence were calculated as described in the Section 2.6.1. In addition, to achieve a fair comparison, the KNIME workflow and machine-learning algorithms were used with the same configurations. The results of all classification algorithms indicate that NeRNA-based negative human miRNA sequences have more significant enhancement in accuracy (Figure 3.1.4).

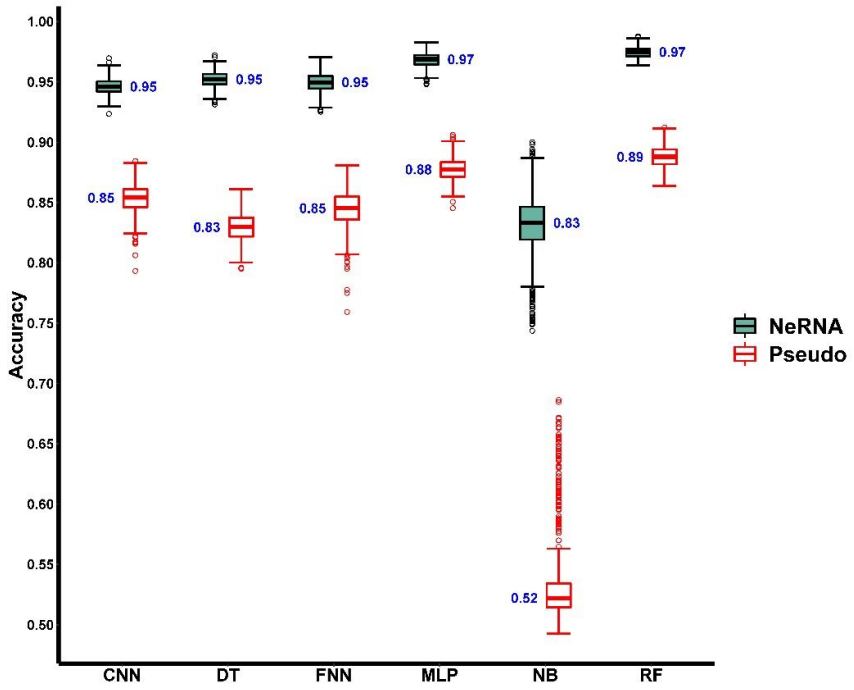

Figure 3.1.4 Box Plots of Classifier Accuracy Scores. Trained with NeRNA and Pseudo miRNA Negative Sequences. Using Various ML Models (CNN, DT, FNN, MLP, NB, RF) with miRNA Hairpin Sequences as Positive Data.

In order to test the suitability of the NeRNA-derived classification algorithm for species, species-specific analyses were performed using miRNA precursor sequences from ten different species [73]. As illustrated in Figure 3.1.5, an average accuracy rate of 88% was observed for all species. Notably, the algorithm achieved the highest accuracy of 98% for the Oreochromis species. Similar to the previous results, the Random Forest exhibited the highest performance among all classifiers, followed by the MLP and DT classifiers. By contrast, the Naive Bayes classifier demonstrated the lowest performance. To investigate this comparison further, many methods that encompassed the same species were used. These algorithms include miRNAFinder [107], miPred [108], microPred [109], plantmiRNAPred [110], triplet-SVM [111] (with accuracy results directly sourced

from relevant papers), and NeRNA (Table 3.1.1). Notably, NeRNA exhibits superior performance to all other software, with the exception of the plantMiRNAPred algorithm for the glycine max species.

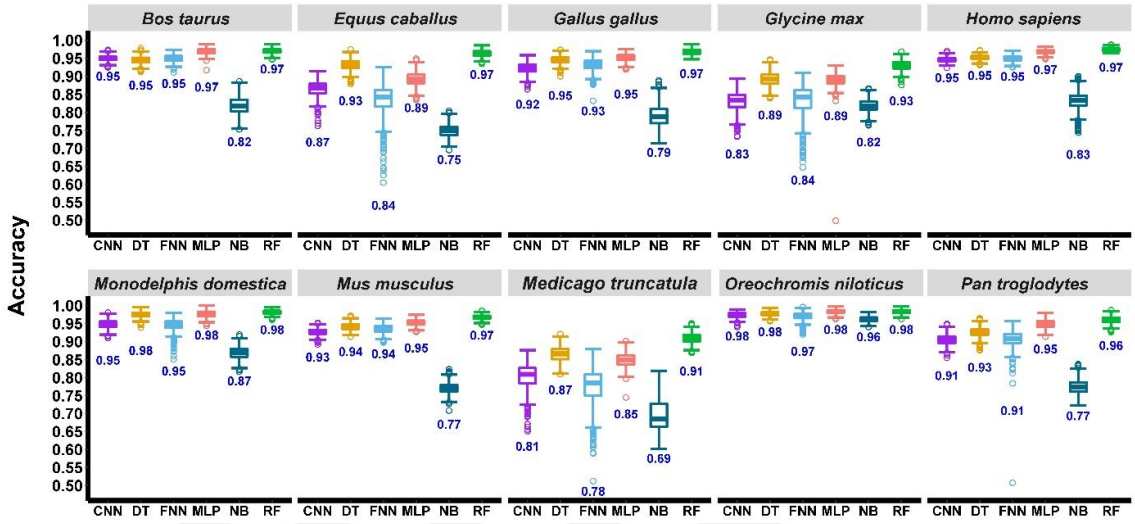

Figure 3.1.5 Species-Specific Performance Comparison of NeRNA. With CNN, DT, FNN, MLP, NB, and RF Classifiers, with Accuracy Scores on the y-axis and Classifiers on the x-axis, showing Q2 (median) percentiles for each species.

Table 3.1.1 Comparison of NeRNA Performance with Similar Methods

	miRNAFinder	**NeRNA**	**MicroPred**	**Mipred**	**triplet-SVM**	**plantMiRNAPred**
Glycine max	0.90	0.93	0.91	0.92	0.91	**0.94**
Medicago truncatula	0.79	**0.91**	0.79	0.88	0.85	0.89
Homo sapiens		**0.96**	0.93	0.91	0.93	
Mus musculus		**0.97**	0.64		0.94	

The evaluation of the effect of features on the classification accuracy of circRNAs, lncRNAs, tRNAs, and miRNAs was evaluated using the information gain calculation approach as a metric for feature selection (Table 3.1.2). The investigation revealed a significant difference in the impact of the characteristics of various RNA types. The outcomes indicated that the most significant features for circRNA and lncRNA had a gain of almost 0.05, while tRNA and miRNA had gains of 0.62 and 0.31, respectively. This was nearly ten times higher than the contribution of lncRNAs and circRNAs.

Furthermore, the first five features of lncRNAs are directly x-based features; however, miRNA classification is mostly influenced by z-based features. This contrast not only underscores the differing significance of characteristics in the classification of RNA but also indicates a complex connection between the predictive capabilities of various feature categories. These findings offer a strong basis for improving feature-selection approaches and increasing the accuracy of RNA classification systems.

Table 3.1.2 Top 5 Feature Selection Based on Information Gain Ratio.

circRNA	Feature Name	Information Gain Ratio	tRNA	Feature Name	Information Gain Ratio
	x2_4	0,048967132		z1_1	0,628537077
	y2_4	0,046810599		z2_4	0,575535882
	x1_3	0,034040497		y3_2	0,477680119
	y1_3	0,027981581		y2_4	0,448667498
	x3_3	0,026410328		y2_2	0,446523874
lncRNA	**Feature Name**	**Information Gain Ratio**	**miRNA**	**Feature Name**	**Information Gain Ratio**
	x1_3	0,052049254		z3_4	0,318827839
	x2_4	0,036973116		z1_4	0,253354739
	x3_2	0,031768159		z1_3	0,249983376
	x3_4	0,03120573		z2_3	0,234772535
	x2_1	0,028551702		z3_3	0,198097593

3.2. Discussion

Non-coding RNAs (ncRNAs) play crucial roles in the cell's regulatory networks, affecting processes from gene regulation by miRNAs to protein synthesis by tRNAs, chromosomal modifications by lncRNAs, and miRNA sequestration by circRNAs [112]. Their functions in gene regulation position them as potential targets for therapeutic interventions and as biomarkers for various diseases [113]. However, the application of machine learning (ML) in ncRNA identification has been limited by the lack of consideration for negative samples in training datasets. Positive samples are typically sourced from databases of known ncRNA classes, but the complex relationships between ncRNAs and their targets make comprehensive experimental validation challenging, leading to the reliance on arbitrary negative datasets.

Addressing the shortcomings in negative dataset construction, previous efforts have explored the impacts of ML on miRNA prediction and the quality of these datasets, with random shuffling of known sequences being a common but imperfect approach [67]. By adopting and modifying dynamic 3D graphical representation techniques, our approach allows the transformation of RNA sequence and structure into numerical data

suitable for ML applications that harbor the diverse properties of ncRNAs [92]. This method represents an important step forward in the accurate classification and understanding of ncRNAs, opening new avenues for research and therapeutic development.

This thesis introduces NeRNA, a novel methodology for the generation of high-quality negative sequences through an ab initio approach. By incorporating secondary structures, mathematical representations, and machine-learning features, NeRNA marks a significant advancement in creating negative data for the classification of non-coding RNAs. Its user-friendly implementation as a KNIME workflow, accessible via GitHub, offering significant contribution to the field of bioinformatics and non-coding RNA research.

The exploration of machine learning and deep learning models across different non-coding RNA groups - miRNA, tRNA, lncRNA and circRNA - has underscored the critical role of negative sequence data which enhancing the performance of classification models. Results indicate that smaller sequences, such as miRNA and tRNA, achieve better accuracy than lncRNA and circRNA (Figure 3.1.1). This discrepancy can be attributed to two main factors. First, the overall length of lncRNA and circRNA sequences, which often exceed 1500 nucleotides, increases the reliance on z-based features for machine learning algorithms. Given KNIME's digit number limitation of 15, some z parameters may return as NULL, consequently reducing the accuracy for lncRNA and circRNA classifications. Second, the highly conserved secondary structures of tRNA and miRNA contribute to distinct differences between their negative and original structures, potentially enhancing calculation accuracy. Contrary to expectation, circRNAs, with their unique circular structure, showed slightly lower accuracy scores than lncRNAs, suggesting that the effect of secondary structure is reduced for sequences longer than 1000 nucleotides.

The adaptation of decimal scaling normalization to overcome learning obstacles with CNN and FNN models signifies a notable advancement in processing lncRNA and circRNA datasets (Figure 3.1.2). Moreover, species-specific analysis extends the applicability of NeRNA, demonstrating its capability to accurately classify RNA sequences across a different species (Figure 3.1.5). This superiority of NeRNA over other existing software, with few exceptions, highlights its potential for broad application in RNA research (Table 3.1.1). The evaluation of features' impact on classification accuracy

also provides deep insights into RNA sequence complexity and indicates avenues for refining feature-selection methods to enhance classification results (Table 3.1.2).

Chapter 4

Conclusions and Future Prospects

4.1. Conclusions

This study introduced a novel methodology for generating negative data for RNA sequences. The proposed NeRNA methodology represents a significant advancement in the generation of high-quality negative sequences for the classification of non-coding RNAs. By integrating secondary structures, mathematical representations, and machine-learning features, NeRNA offers a novel approach to address the challenges associated with negative dataset construction in the field of RNA identification. The experimental results from machine learning and deep learning models applied across different ncRNA groups (miRNA, tRNA, lncRNA, and circRNA) underscore the critical importance of incorporating negative sequence data to enhance the classification model performance (Figure 3.1.1 and Figure 3.1.3). The comparison analysis results of the miRNA classification showed that NeRNA generated negative sequences that were superior to those of existing datasets (Figure 3.1.4). Although the method is designed for noncoding RNAs, it can also be applied to coding RNAs as well as pathogenic and/or viral RNA sequences. The ongoing refinement of these tools and their integration into broader research and clinical frameworks will undoubtedly contribute to significant discoveries and improvements in health care.

4.2. Societal Impact and Contribution to Global Sustainability

The NeRNA methodology constitutes a significant advancement in the field of non-coding RNA (ncRNA) research, with important societal implications and contributions to global sustainability. By offering an accessible workflow within the Konstanz Information Miner (KNIME) system, NeRNA democratizes access to advanced bioinformatics tools, empowering researchers worldwide to generate high-quality negative sequences that are critical for improving the accuracy of machine learning models. This accessibility encourages collaboration among scientists and institutions, accelerating the process of drug discovery and targeted therapy development. The improved accuracy of ncRNA classification achieved through NeRNA not only enhances biomedical research outcomes but also contributes to reducing healthcare costs by facilitating the more efficient identification of potential therapeutic targets.

NeRNA embodies the principles of sustainability in the fields of biotechnology and computational biology. By optimizing computational resources through its efficient workflow, NeRNA minimizes energy consumption and waste. This aligns with the goal of eco-conscious research. NeRNA's reduction in resource consumption contributes to a more sustainable research environment, ensuring that advancements in bioinformatics and AI (Artificial Intelligence) are achieved responsibly. In addition, NeRNA plays a critical role in refining machine learning models for ncRNA classification, demonstrating the ethical deployment of AI technologies in healthcare. Responsible AI applications, as exemplified by NeRNA, prioritize transparency, fairness, and accountability, which are essential for building trust in AI-driven healthcare innovations.

The open-source nature of NeRNA has the potential to promote knowledge-sharing and cooperative research, thereby enhancing the reproducibility and transparency of bioinformatics studies on a global scale. This open-access approach to advanced bioinformatics tools empowers researchers, educators, and students to engage in cutting-edge research, thereby democratizing the benefits of scientific progress. The impact of NeRNA extends beyond the scientific community, influencing healthcare innovations and sustainable research practices that align with global sustainability goals. As society faces increasingly complex healthcare challenges and demands for responsible technological innovation, NeRNA serves as a critical step forward in leveraging AI and bioinformatics for the well-being and sustainable development of society.

4.3. Future Prospects

The use of NeRNA for generating high-quality negative sequences for non-coding RNA (ncRNA) classification heralds several pathways for future research. Its efficacy in addressing negative data generation challenges and enhancing machine learning (ML) models for ncRNA classification underscores its potential for broader applications. Future endeavors could aim at broadening NeRNA's scope to encompass more ncRNA species and exploring its relevance in genomic studies beyond the current scope. Moreover, integrating NeRNA into comprehensive bioinformatics workflows could revolutionize the automated discovery and functional annotation of ncRNAs across various organisms.

The application of ML and deep learning in ncRNA classification, especially through NeRNA, shows promise. Future research should refine these models to tackle the specific challenges of longer ncRNAs, such as lncRNAs and circRNAs, where digit limitations have impacted accuracy. Enhancing model precision and efficiency might be achieved by incorporating innovative feature selection methods, optimization techniques, or by broadening the definition of x, y, z-based features.

NeRNA offers a novel approach to generate high-quality negative sequences, while its application can be extended beyond the current ncRNA groups studied (miRNA, tRNA, lncRNA and circRNA). It has the potential to produce quality negative data for all RNA, DNA, and protein sequences, offering deeper insights into gene regulation and expression complexities. Furthermore, integrating the NeRNA workflow with other bioinformatics tools and databases can increase its utility and accessibility for researchers.

It can also be extended to the calculation of secondary structures, which is one of the main topics of NeRNA. The RNAfold software utilized in this thesis faces limitations with sequences over 30,000 nucleotides and struggles to accurately generate secondary structures for modified molecules, such as pseudouridine in tRNA. Therefore, the development of a new secondary structure calculation algorithm in future studies will make a great contribution in terms of both ease and accuracy of the process.

BIBLIOGRAPHY

[1] P. E. Saw, X. Xu, J. Chen, and E.-W. Song, "Non-coding RNAs: the new central dogma of cancer biology," *Sci China Life Sci*, vol. 64, no. 1, pp. 22–50, Jan. 2021, doi: 10.1007/s11427-020-1700-9.

[2] J. Wells, "Not Junk After All: Non-Protein-Coding DNA Carries Extensive Biological Information," in *Biological Information*, WORLD SCIENTIFIC, Jul. 2013, pp. 210–231. doi: 10.1142/9789814508728_0009.

[3] P. Carninci *et al.*, "The transcriptional landscape of the mammalian genome," *Science*, vol. 309, no. 5740, pp. 1559–1563, Sep. 2005, doi: 10.1126/SCIENCE.1112014.

[4] "Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project," *Nature*, vol. 447, no. 7146, pp. 799–816, Jun. 2007, doi: 10.1038/nature05874.

[5] J. S. Mattick, "Non-coding RNAs: the architects of eukaryotic complexity," *EMBO Rep*, vol. 2, no. 11, p. 986, Nov. 2001, doi: 10.1093/EMBO-REPORTS/KVE230.

[6] L. Sun, X. Chen, and Z. Jin, "Emerging roles of non-coding RNAs in retinal diseases: A review," *Clin Exp Ophthalmol*, vol. 48, no. 8, pp. 1085–1101, Nov. 2020, doi: 10.1111/ceo.13806.

[7] A. F. Gabriel, M. C. Costa, and F. J. Enguita, "Interactions Among Regulatory Non-coding RNAs Involved in Cardiovascular Diseases," 2020, pp. 79–104. doi: 10.1007/978-981-15-1671-9_4.

[8] Shiv Swaroop and Thangminlal Vaiphei S, "Potential Roles of Long Non-Coding RNAs (lncRNAs) in Stress Response Regulation," *International Journal of Research in Pharmaceutical Sciences*, vol. 11, no. SPL4, pp. 2385–2389, Dec. 2020, doi: 10.26452/ijrps.v11iSPL4.4482.

[9] J. J. Chan and Y. Tay, "Noncoding RNA:RNA Regulatory Networks in Cancer," *Int J Mol Sci*, vol. 19, no. 5, May 2018, doi: 10.3390/IJMS19051310.

[10] A. A. Bhat *et al.*, "Role of non-coding RNA networks in leukemia progression, metastasis and drug resistance," *Mol Cancer*, vol. 19, no. 1, pp. 1–21, Mar. 2020, doi: 10.1186/S12943-020-01175-9/FIGURES/4.

[11] M. Esteller, "Non-coding RNAs in human disease," *Nat Rev Genet*, vol. 12, no. 12, pp. 861–874, Dec. 2011, doi: 10.1038/nrg3074.

[12] Y. Zhang *et al.*, "Circular intronic long noncoding RNAs," *Mol Cell*, vol. 51, no. 6, pp. 792–806, Sep. 2013, doi: 10.1016/J.MOLCEL.2013.08.017.

[13] S. Guil and M. Esteller, "RNA-RNA interactions in gene regulation: the coding and noncoding players," *Trends Biochem Sci*, vol. 40, no. 5, pp. 248–256, May 2015, doi: 10.1016/J.TIBS.2015.03.001.

[14] K. Saliminejad, H. R. Khorram Khorshid, S. Soleymani Fard, and S. H. Ghaffari, "An overview of microRNAs: Biology, functions, therapeutics, and analysis methods," *J Cell Physiol*, vol. 234, no. 5, pp. 5451–5465, May 2019, doi: 10.1002/JCP.27486.

[15] B. Wightman, I. Ha, and G. Ruvkun, "Posttranscriptional regulation of the heterochronic gene lin-14 by lin-4 mediates temporal pattern formation in C. elegans," *Cell*, vol. 75, no. 5, pp. 855–862, Dec. 1993, doi: 10.1016/0092-8674(93)90530-4.

[16] D. P. Bartel, “MicroRNAs: target recognition and regulatory functions,” *Cell*, vol. 136, no. 2, pp. 215–233, Jan. 2009, doi: 10.1016/J.CELL.2009.01.002.
[17] R. C. Friedman, K. K. H. Farh, C. B. Burge, and D. P. Bartel, “Most mammalian mRNAs are conserved targets of microRNAs,” *Genome Res*, vol. 19, no. 1, p. 92, Jan. 2009, doi: 10.1101/GR.082701.108.
[18] W. Filipowicz, S. N. Bhattacharyya, and N. Sonenberg, “Mechanisms of post-transcriptional regulation by microRNAs: are the answers in sight?,” *Nature Reviews Genetics 2008 9:2*, vol. 9, no. 2, pp. 102–114, Feb. 2008, doi: 10.1038/nrg2290.
[19] A. Eulalio, E. Huntzinger, and E. Izaurralde, “Getting to the root of miRNA-mediated gene silencing,” *Cell*, vol. 132, no. 1, pp. 9–14, Jan. 2008, doi: 10.1016/J.CELL.2007.12.024.
[20] M. Angulo, E. Lecuona, and J. I. Sznajder, “Role of MicroRNAs in Lung Disease,” *Archivos de Bronconeumología (English Edition)*, vol. 48, no. 9, pp. 325–330, Sep. 2012, doi: 10.1016/j.arbr.2012.06.015.
[21] E. Javandoost, E. Firoozi-Majd, H. Rostamian, M. Khakpoor- Koosheh, and H. R. Mirzaei, “Role of microRNAs in Chronic Lymphocytic Leukemia Pathogenesis,” *Curr Med Chem*, vol. 27, no. 2, pp. 282–297, Feb. 2020, doi: 10.2174/0929867326666190911114842.
[22] C. Yapijakis, “Regulatory Role of MicroRNAs in Brain Development and Function,” 2020, pp. 237–247. doi: 10.1007/978-3-030-32633-3_32.
[23] Q. Wei, Q. Mi, and Z. Dong, “The regulation and function of micrornas in kidney diseases,” *IUBMB Life*, vol. 65, no. 7, pp. 602–614, Jul. 2013, doi: 10.1002/iub.1174.
[24] V. Tubita *et al.*, “Role of microRNAs in inflammatory upper airway diseases,” *Allergy*, vol. 76, no. 7, pp. 1967–1980, Jul. 2021, doi: 10.1111/all.14706.
[25] D. M. Cerqueira, M. Tayeb, and J. Ho, “MicroRNAs in kidney development and disease,” *JCI Insight*, vol. 7, no. 9, May 2022, doi: 10.1172/jci.insight.158277.
[26] K. U. Tüfekci, M. G. Öner, R. L. J. Meuwissen, and Ş. Genç, “The Role of MicroRNAs in Human Diseases,” 2014, pp. 33–50. doi: 10.1007/978-1-62703-748-8_3.
[27] S. Mukhadi, R. Hull, Z. Mbita, and Z. Dlamini, “The Role of MicroRNAs in Kidney Disease,” *Noncoding RNA*, vol. 1, no. 3, pp. 192–221, Nov. 2015, doi: 10.3390/ncrna1030192.
[28] A. Ceribelli, B. Yao, P. R. Dominguez-Gutierrez, M. A. Nahid, M. Satoh, and E. K. Chan, “MicroRNAs in systemic rheumatic diseases,” *Arthritis Res Ther*, vol. 13, no. 4, p. 229, 2011, doi: 10.1186/ar3377.
[29] D. M. Cerqueira, M. Tayeb, and J. Ho, “MicroRNAs in kidney development and disease,” *JCI Insight*, vol. 7, no. 9, May 2022, doi: 10.1172/jci.insight.158277.
[30] M. Yang and J. Mattes, “Discovery, biology and therapeutic potential of RNA interference, microRNA and antagomirs,” *Pharmacol Ther*, vol. 117, no. 1, pp. 94–104, Jan. 2008, doi: 10.1016/j.pharmthera.2007.08.004.
[31] M. Yang and J. Mattes, “Discovery, biology and therapeutic potential of RNA interference, microRNA and antagomirs,” *Pharmacol Ther*, vol. 117, no. 1, pp. 94–104, Jan. 2008, doi: 10.1016/j.pharmthera.2007.08.004.
[32] J. G and D. Moras, “Transfer RNA,” in *Oxford Handbook of Nucleic Acid Structure*, Oxford University PressOxford, 1999, pp. 603–652. doi: 10.1093/oso/9780198500384.003.0019.
[33] T. Pan, “Modifications and functional genomics of human transfer RNA,” *Cell Res*, vol. 28, no. 4, pp. 395–404, Apr. 2018, doi: 10.1038/s41422-018-0013-y.

[34] M. D. Berg and C. J. Brandl, “Transfer RNAs: diversity in form and function,” *RNA Biol*, vol. 18, no. 3, p. 316, 2021, doi: 10.1080/15476286.2020.1809197.
[35] T. Yokogawa *et al.*, “A novel cloverleaf structure found in mammalian mitochondrial tRNA ^{Ser} (UCN),” *Nucleic Acids Res*, vol. 19, no. 22, pp. 6101–6105, 1991, doi: 10.1093/nar/19.22.6101.
[36] J. C. Biro, “The concept of RNA-assisted protein folding: the role of tRNA,” *Theor Biol Med Model*, vol. 9, no. 1, p. 10, Dec. 2012, doi: 10.1186/1742-4682-9-10.
[37] B. Ruan *et al.*, “Genomics and the evolution of aminoacyl-tRNA synthesis.,” *Acta Biochim Pol*, vol. 48, no. 2, pp. 313–321, Jun. 2001, doi: 10.18388/abp.2001_3917.
[38] M. Chery and L. Drouard, “Plant tRNA functions beyond their major role in translation,” *J Exp Bot*, vol. 74, no. 7, pp. 2352–2363, Apr. 2023, doi: 10.1093/jxb/erac483.
[39] E. A. Orellana, E. Siegal, and R. I. Gregory, “tRNA dysregulation and disease,” *Nat Rev Genet*, vol. 23, no. 11, pp. 651–664, Nov. 2022, doi: 10.1038/s41576-022-00501-9.
[40] E. Borek, “Transfer RNA and Its By-Products as Tumor Markers,” in *Cancer Markers*, Totowa, NJ: Humana Press, 1980, pp. 445–462. doi: 10.1007/978-1-4612-6117-9_16.
[41] H. L. Sanger, G. Klotz, D. Riesner, H. J. Gross, and A. K. Kleinschmidt, “Viroids are single-stranded covalently closed circular RNA molecules existing as highly base-paired rod-like structures,” *Proc Natl Acad Sci U S A*, vol. 73, no. 11, pp. 3852–3856, 1976, doi: 10.1073/PNAS.73.11.3852.
[42] J. U. Guo, V. Agarwal, H. Guo, and D. P. Bartel, “Expanded identification and characterization of mammalian circular RNAs,” *Genome Biol*, vol. 15, no. 7, Jul. 2014, doi: 10.1186/S13059-014-0409-Z.
[43] T. B. Hansen *et al.*, “Natural RNA circles function as efficient microRNA sponges,” *Nature*, vol. 495, no. 7441, pp. 384–388, Mar. 2013, doi: 10.1038/NATURE11993.
[44] C.-Y. Yu and H.-C. Kuo, “The emerging roles and functions of circular RNAs and their generation,” *J Biomed Sci*, vol. 26, no. 1, p. 29, Dec. 2019, doi: 10.1186/s12929-019-0523-z.
[45] Y. Zeng *et al.*, “The biogenesis, function and clinical significance of circular RNAs in breast cancer,” *Cancer Biol Med*, vol. 18, no., pp. 0–0, 2021, doi: 10.20892/j.issn.2095-3941.2020.0485.
[46] M. N. Abbas *et al.*, “The Potential Biological Roles of Circular RNAs in the Immune Systems of Insects to Pathogen Invasion,” *Genes (Basel)*, vol. 14, no. 4, p. 895, Apr. 2023, doi: 10.3390/genes14040895.
[47] Y. Tang, T. Zhou, X. Yu, Z. Xue, and N. Shen, “The role of long non-coding RNAs in rheumatic diseases,” *Nat Rev Rheumatol*, vol. 13, no. 11, pp. 657–669, Nov. 2017, doi: 10.1038/nrrheum.2017.162.
[48] L. Statello, C.-J. Guo, L.-L. Chen, and M. Huarte, “Gene regulation by long non-coding RNAs and its biological functions,” *Nat Rev Mol Cell Biol*, vol. 22, no. 2, pp. 96–118, Feb. 2021, doi: 10.1038/s41580-020-00315-9.
[49] P.-O. Angrand, C. Vennin, X. Le Bourhis, and E. Adriaenssens, “The role of long non-coding RNAs in genome formatting and expression,” *Front Genet*, vol. 6, Apr. 2015, doi: 10.3389/fgene.2015.00165.

[50] V. Mouraviev *et al.*, “Clinical prospects of long noncoding RNAs as novel biomarkers and therapeutic targets in prostate cancer,” *Prostate Cancer Prostatic Dis*, vol. 19, no. 1, pp. 14–20, Mar. 2016, doi: 10.1038/pcan.2015.48.
[51] Q. Chen, C. Wei, Z. Wang, and M. Sun, “Long non-coding RNAs in anti-cancer drug resistance,” *Oncotarget*, vol. 8, no. 1, pp. 1925–1936, Jan. 2017, doi: 10.18632/oncotarget.12461.
[52] H. Satam *et al.*, “Next-Generation Sequencing Technology: Current Trends and Advancements,” *Biology (Basel)*, vol. 12, no. 7, Jul. 2023, doi: 10.3390/BIOLOGY12070997.
[53] Y. Kim and M. Lee, “Deep Learning Approaches for lncRNA-Mediated Mechanisms: A Comprehensive Review of Recent Developments,” *International Journal of Molecular Sciences 2023, Vol. 24, Page 10299*, vol. 24, no. 12, p. 10299, Jun. 2023, doi: 10.3390/IJMS241210299.
[54] M. Awad and R. Khanna, “Machine Learning,” *Efficient Learning Machines*, pp. 1–18, 2015, doi: 10.1007/978-1-4302-5990-9_1.
[55] C. Caudai *et al.*, “AI applications in functional genomics,” *Comput Struct Biotechnol J*, vol. 19, pp. 5762–5790, Jan. 2021, doi: 10.1016/J.CSBJ.2021.10.009.
[56] S. F. Weng, J. Reps, J. Kai, J. M. Garibaldi, and N. Qureshi, “Can machine-learning improve cardiovascular risk prediction using routine clinical data?,” *PLoS One*, vol. 12, no. 4, p. e0174944, Apr. 2017, doi: 10.1371/JOURNAL.PONE.0174944.
[57] M. N. Asim, M. A. Ibrahim, M. Imran Malik, A. Dengel, and S. Ahmed, “Advances in Computational Methodologies for Classification and Sub-Cellular Locality Prediction of Non-Coding RNAs,” *Int J Mol Sci*, vol. 22, no. 16, Aug. 2021, doi: 10.3390/IJMS22168719.
[58] A. El Allali, Z. Elhamraoui, and R. Daoud, “Machine learning applications in RNA modification sites prediction,” *Comput Struct Biotechnol J*, vol. 19, pp. 5510–5524, Jan. 2021, doi: 10.1016/J.CSBJ.2021.09.025.
[59] K. Y. Yip, C. Cheng, and M. Gerstein, “Machine learning and genome annotation: a match meant to be?,” *Genome Biol*, vol. 14, no. 5, p. 205, May 2013, doi: 10.1186/GB-2013-14-5-205.
[60] S. Huang, C. A. I. Nianguang, P. Penzuti Pacheco, S. Narandes, Y. Wang, and X. U. Wayne, “Applications of Support Vector Machine (SVM) Learning in Cancer Genomics,” *Cancer Genomics Proteomics*, vol. 15, no. 1, p. 41, Jan. 2018, doi: 10.21873/CGP.20063.
[61] P. Larrañaga *et al.*, “Machine learning in bioinformatics,” *Brief Bioinform*, vol. 7, no. 1, pp. 86–112, Mar. 2006, doi: 10.1093/BIB/BBK007.
[62] M. D. Saçar and J. Allmer, “Machine learning methods for microRNA gene prediction,” *Methods in Molecular Biology*, vol. 1107, 2014, doi: 10.1007/978-1-62703-748-8_10.
[63] K. Ghosh, C. Bellinger, R. Corizzo, P. Branco, B. Krawczyk, and N. Japkowicz, “The class imbalance problem in deep learning,” *Mach Learn*, pp. 1–57, Dec. 2022, doi: 10.1007/S10994-022-06268-8/FIGURES/27.
[64] S. A. Hicks *et al.*, “On evaluation metrics for medical applications of artificial intelligence,” *Scientific Reports 2022 12:1*, vol. 12, no. 1, pp. 1–9, Apr. 2022, doi: 10.1038/s41598-022-09954-8.
[65] M. Ali, A. Dewan, A. K. Sahu, and M. M. Taye, “Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future

Directions," *Computers 2023, Vol. 12, Page 91*, vol. 12, no. 5, p. 91, Apr. 2023, doi: 10.3390/COMPUTERS12050091.
[66] D. Singh and J. Roy, "A large-scale benchmark study of tools for the classification of protein-coding and non-coding RNAs," *Nucleic Acids Res*, vol. 50, no. 21, p. 12094, Nov. 2022, doi: 10.1093/NAR/GKAC1092.
[67] J. Caballero, A. F. A. Smit, L. Hood, and G. Glusman, "Realistic artificial DNA sequences as negative controls for computational genomics," *Nucleic Acids Res*, vol. 42, no. 12, Jul. 2014, doi: 10.1093/NAR/GKU356.
[68] M. R. Berthold *et al.*, "KNIME: The Konstanz Information Miner," in *SIGKDD Explorations*, vol. 11, no. 1, 2008, pp. 319–326. doi: 10.1007/978-3-540-78246-9_38.
[69] R. Lorenz *et al.*, "ViennaRNA Package 2.0," *Algorithms for Molecular Biology*, vol. 6, no. 1, pp. 1–14, Nov. 2011, doi: 10.1186/1748-7188-6-26/TABLES/2.
[70] D. Charif and J. R. Lobry, "SeqinR 1.0-2: A Contributed Package to the R Project for Statistical Computing Devoted to Biological Sequences Retrieval and Analysis," pp. 207–232, 2007, doi: 10.1007/978-3-540-35306-5_10.
[71] F. Chollet and others, "Keras." 2015.
[72] Martín~Abadi *et al.*, " TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems." 2015. [Online]. Available: https://www.tensorflow.org/
[73] S. Griffiths-Jones, "miRBase: microRNA Sequences and Annotation," *Curr Protoc Bioinformatics*, vol. 29, no. 1, Mar. 2010, doi: 10.1002/0471250953.bi1209s29.
[74] P. J. Volders *et al.*, "LNCipedia 5: towards a reference set of human long non-coding RNAs," *Nucleic Acids Res*, vol. 47, no. D1, pp. D135–D139, Jan. 2019, doi: 10.1093/NAR/GKY1031.
[75] P. Glažar, P. Papavasileiou, and N. Rajewsky, "circBase: a database for circular RNAs," *RNA*, vol. 20, no. 11, pp. 1666–1670, Sep. 2014, doi: 10.1261/RNA.043687.113.
[76] M. P. Sajek, T. Woźniak, M. Sprinzl, J. Jaruzelska, and J. Barciszewski, "T-psi-C: user friendly database of tRNA sequences and structures," *Nucleic Acids Res*, vol. 48, no. D1, pp. D256–D260, Jan. 2020, doi: 10.1093/NAR/GKZ922.
[77] M. E. Nebel, "Combinatorial Properties of RNA Secondary Structures," *Journal of Computational Biology*, vol. 9, no. 3, pp. 541–573, Jun. 2002, doi: 10.1089/106652702760138628.
[78] S. Griffiths-Jones, "miRBase: microRNA sequences and annotation.," *Current protocols in bioinformatics / editoral board, Andreas D. Baxevanis ... [et al.]*, vol. Chapter 12, p. Unit 12.9.1-10, Mar. 2010, doi: 10.1002/0471250953.bi1209s29.
[79] P. J. Volders *et al.*, "LNCipedia 5: towards a reference set of human long non-coding RNAs," *Nucleic Acids Res*, vol. 47, no. D1, pp. D135–D139, Jan. 2019, doi: 10.1093/NAR/GKY1031.
[80] P. Glažar, P. Papavasileiou, and N. Rajewsky, "circBase: a database for circular RNAs.," *RNA*, vol. 20, no. 11, pp. 1666–70, Nov. 2014, doi: 10.1261/rna.043687.113.
[81] M. P. Sajek, T. Woźniak, M. Sprinzl, J. Jaruzelska, and J. Barciszewski, "T-psi-C: user friendly database of tRNA sequences and structures," *Nucleic Acids Res*, vol. 48, no. D1, pp. D256–D260, Jan. 2020, doi: 10.1093/NAR/GKZ922.
[82] J. C. Wu, D. P. Gardner, S. Ozer, R. R. Gutell, and P. Ren, "Correlation of RNA Secondary Structure Statistics with Thermodynamic Stability and Applications to

Folding," *J Mol Biol*, vol. 391, no. 4, pp. 769–783, Aug. 2009, doi: 10.1016/j.jmb.2009.06.036.
[83] M. Tacker, P. F. Stadler, E. G. Bornberg-Bauer, I. L. Hofacker, and P. Schuster, "Algorithm independent properties of RNA secondary structure predictions," *European Biophysics Journal*, vol. 25, no. 2, pp. 115–130, Dec. 1996, doi: 10.1007/s002490050023.
[84] K. Sato, M. Akiyama, and Y. Sakakibara, "RNA secondary structure prediction using deep learning with thermodynamic integration," *Nat Commun*, vol. 12, no. 1, p. 941, Feb. 2021, doi: 10.1038/s41467-021-21194-4.
[85] H. Yonemoto, K. Asai, and M. Hamada, "A semi-supervised learning approach for RNA secondary structure prediction," *Comput Biol Chem*, vol. 57, pp. 72–79, Aug. 2015, doi: 10.1016/j.compbiolchem.2015.02.002.
[86] H. Shi and X. Jing, "Efficient Generation of RNA Secondary Structure Prediction Algorithm Under PAR Framework," *Front Plant Sci*, vol. 12, Jan. 2022, doi: 10.3389/fpls.2021.830042.
[87] C. Liu, Z. Ji, and Y. Hong, "A Parallel Shuffled Frog Leaping Algorithm Based on Stem Regions Combinatorial Optimization for RNA Secondary Structure Prediction," in *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, Paris, France: Atlantis Press, 2013. doi: 10.2991/iccsee.2013.320.
[88] Q. Zhao, Z. Zhao, X. Fan, Z. Yuan, Q. Mao, and Y. Yao, "Review of machine learning methods for RNA secondary structure prediction," *PLoS Comput Biol*, vol. 17, no. 8, p. e1009291, Aug. 2021, doi: 10.1371/journal.pcbi.1009291.
[89] M. F. Sloma and D. H. Mathews, "Improving RNA Secondary Structure Prediction with Structure Mapping Data," 2015, pp. 91–114. doi: 10.1016/bs.mie.2014.10.053.
[90] T. J. Macke, D. J. Ecker, R. R. Gutell, D. Gautheret, D. A. Case, and R. Sampath, "RNAMotif, an RNA secondary structure definition and search algorithm," *Nucleic Acids Res*, vol. 29, no. 22, pp. 4724–4735, Nov. 2001, doi: 10.1093/NAR/29.22.4724.
[91] E. Mattei, G. Ausiello, F. Ferrè, and M. Helmer-Citterich, "A novel approach to represent and compare RNA secondary structures," *Nucleic Acids Res*, vol. 42, no. 10, p. 6146, Jun. 2014, doi: 10.1093/NAR/GKU283.
[92] Y. Zhang *et al.*, "A dynamic 3D graphical representation for RNA structure analysis and its application in non-coding RNA classification," *PLoS One*, vol. 11, no. 5, pp. 1–15, 2016, doi: 10.1371/journal.pone.0152238.
[93] M. D. Saçar Demirci and J. Allmer, "izMiR ab initio MicroRNA Analysis," vol. 1, 2019, doi: 10.17632/MGH5R9WNY7.1.
[94] X. Chen, M. Wang, and H. Zhang, "The use of classification trees for bioinformatics," *Wiley Interdiscip Rev Data Min Knowl Discov*, vol. 1, no. 1, p. 55, Jan. 2011, doi: 10.1002/WIDM.14.
[95] S. Min, B. Lee, and S. Yoon, "Deep learning in bioinformatics," *Brief Bioinform*, vol. 18, no. 5, pp. 851–869, Sep. 2017, doi: 10.1093/BIB/BBW068.
[96] M. D. Saçar and J. Allmer, "Data mining for microrna gene prediction: On the impact of class imbalance and feature number for microrna gene prediction," *2013 8th International Symposium on Health Informatics and Bioinformatics, HIBIT 2013*, 2013, doi: 10.1109/HIBIT.2013.6661685.
[97] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal Of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2011, doi: 10.1613/jair.953.

[98] Q. S. Xu and Y. Z. Liang, “Monte Carlo cross validation,” *Chemometrics and Intelligent Laboratory Systems*, vol. 56, no. 1, pp. 1–11, Apr. 2001, doi: 10.1016/S0169-7439(00)00122-2.
[99] Jude Chukwura Obi, “A comparative study of several classification metrics and their performances on data,” *World Journal of Advanced Engineering Technology and Sciences*, vol. 8, no. 1, pp. 308–314, Feb. 2023, doi: 10.30574/wjaets.2023.8.1.0054.
[100] S. Gopal, K. Patro, and K. Kumar Sahu, “Normalization: A Preprocessing Stage,” *IARJSET*, pp. 20–22, Mar. 2015, doi: 10.17148/iarjset.2015.2305.
[101] R. A. M. Villanueva and Z. J. Chen, “ggplot2: Elegant Graphics for Data Analysis (2nd ed.),” *https://doi.org/10.1080/15366367.2019.1565254*, vol. 17, no. 3, pp. 160–167, Jul. 2019, doi: 10.1080/15366367.2019.1565254.
[102] A. Kassambara, “ggpubr: ‘ggplot2’ Based Publication Ready Plots.” 2023. [Online]. Available: https://rpkgs.datanovia.com/ggpubr/
[103] X. Robin *et al.*, “pROC: an open-source package for R and S+ to analyze and compare ROC curves,” *BMC Bioinformatics*, vol. 12, no. 1, p. 77, Dec. 2011, doi: 10.1186/1471-2105-12-77.
[104] A. P. Bradley, “The use of the area under the ROC curve in the evaluation of machine learning algorithms,” *Pattern Recognit*, vol. 30, no. 7, pp. 1145–1159, Jul. 1997, doi: 10.1016/S0031-3203(96)00142-2.
[105] R. S, A. S. Bharadwaj, D. S K, M. S. Khadabadi, and A. Jayaprakash, “Digital Implementation of the Softmax Activation Function and the Inverse Softmax Function,” in *2022 4th International Conference on Circuits, Control, Communication and Computing (I4C)*, IEEE, Dec. 2022, pp. 64–67. doi: 10.1109/I4C57141.2022.10057747.
[106] M. D. Saçar Demirci, M. Yousef, and J. Allmer, “Computational Prediction of Functional MicroRNA-mRNA Interactions,” *Methods Mol Biol*, vol. 1912, pp. 175–196, 2019, doi: 10.1007/978-1-4939-8982-9_7.
[107] S. Lokuge, S. Jayasundara, P. Ihalagedara, I. Kahanda, and D. Herath, “miRNAFinder: A comprehensive web resource for plant Pre-microRNA classification,” *Biosystems*, vol. 215–216, p. 104662, Jun. 2022, doi: 10.1016/J.BIOSYSTEMS.2022.104662.
[108] P. Jiang, H. Wu, W. Wang, W. Ma, X. Sun, and Z. Lu, “MiPred: classification of real and pseudo microRNA precursors using random forest prediction model with combined features,” *Nucleic Acids Res*, vol. 35, no. Web Server issue, p. W339, Jul. 2007, doi: 10.1093/NAR/GKM368.
[109] R. Batuwita and V. Palade, “microPred: effective classification of pre-miRNAs for human miRNA gene prediction,” *Bioinformatics*, vol. 25, no. 8, pp. 989–995, Apr. 2009, doi: 10.1093/BIOINFORMATICS/BTP107.
[110] P. Xuan, M. Guo, X. Liu, Y. Huang, W. Li, and Y. Huang, “PlantMiRNAPred: efficient classification of real and pseudo plant pre-miRNAs,” *Bioinformatics*, vol. 27, no. 10, pp. 1368–1376, May 2011, doi: 10.1093/BIOINFORMATICS/BTR153.
[111] C. Xue, F. Li, T. He, G.-P. Liu, Y. Li, and X. Zhang, “Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine,” *BMC Bioinformatics*, vol. 6, no. 1, p. 310, Dec. 2005, doi: 10.1186/1471-2105-6-310.
[112] X. D. Fu, “Non-coding RNA: a new frontier in regulatory biology,” *Natl Sci Rev*, vol. 1, no. 2, pp. 190–204, Jun. 2014, doi: 10.1093/NSR/NWU008.

[113] U. Ala, “Competing Endogenous RNAs, Non-Coding RNAs and Diseases: An Intertwined Story,” *Cells*, vol. 9, no. 7. 2020. doi: 10.3390/cells9071574.

APPENDIX

The list of 101 organisms was compiled from the T-Psi-C Database.

Aedes albopictus, Aeropyrum pernix, Agrobacterium radiobacter, Aquifex aeolicus, Archaeoglobus fulgidus, Ascaris suum, Asterias amurensis, Avian myeloblastosis virus, Azospirillum lipoferum, Bacillus subtilis, Bombyx mori, Bos taurus, Brassica napus, Caenorhabditis elegans, Candida albicans, Candida cylindracea, Caulobacter crescentus CB15, Chlamydomonas reinhardtii, Codium fragile, Cucumis sativus, Desulfitobacterium hafniense, Dictyostelium discoideum, Didelphis virginiana, Drosophila melanogaster, Enterobacteria phage T4, Enterobacteria phage T5, Escherichia coli, Euglena gracilis, Euphausia superba, Gallus gallus, Geobacillus kaustophilus, Geobacillus stearothermophilus, Glycine max, Halobacterium salinarum, Halococcus morrhuae, Halocynthia roretzi, Haloferax volcanii, Hoaloarcula marismortui, Homo sapiens, Hordeum vulgare, Lactococcus lactis, Leishmania tarentolae, Loligo bleekeri, Lupinus albus, Lupinus luteus, Mesocentrotus nudus, Mesocricetus auratus, Methanobacterium thermaggregans, Methanocaldococcus jannaschii, Methanopyrus kandleri, Methanosarcina barkeri, Methanosarcina mazei, Methanothermobacter thermautotrophicus, Moloney murine leukemia virus, Mus musculus, Mycobacterium smegmatis, Mycoplasma capricolum, Mycoplasma mycoides, Nanoarchaeum equitans, Neurospora crassa, Nicotiana rustica, Nicotiana tabacum, Oceanobacillus iheyensis, Oenothera sp., Oryctolagus cuniculus, Ovis aries, Phaseolus vulgaris, Pichia jadinii, Pisum sativum, Pseudomonas aureginosa, Pyrococcus horikoshii, Rattus norvegicus, Rhodospirillum rubrum, Saccharomyces cerevisiae, Salmo salar, Salmonella typhimurium, Scenedesmus obliquus, Schizosaccharomyces pombe, Sinorhizobium meliloti, Solanum tuberosum, Spinacia oleracea, Spiroplasma citri, Staphylococcus aureus, Staphylococcus epidermidis, Streptomyces coelicolor A3(2), Streptomyces griseus, Sulfolobus acidocaldarius, Synechococcus elongatus PCC 6301, Synechococcus sp. PCC 7002, Synechocystis sp., Tetrahymena pyriformis, Tetrahymena thermophila, Thermoplasma acidophilum, Thermotoga maritima, Thermus thermophilus,

Toxoplasma gondii, Triticum aestivum, Trypanosoma brucei, Vibrio cholerae C6706, Xenopus laevis, Zea mays.

Machine Learning and Deep Learning Algorithms parameters on KNIME

Algorithm Name	Parameters		
Decision Tree (DT)	**Quality measure**		Gini index
	Min number record per node		2
	Number record to store for view		100
	Number threads		4
Random Forest (RF)	**Split criterion**		Information gain ratio
	Number of models		100
Naive Bayes (NB)	**Default Probability**		0.0001
	Minimum standard deviation		0.0001
	Threshold standard deviation		0.0
	Maximum number o unique nominal values per attribute		20
Multilayer Perceptron (MLP)	**Maximum number of iterations**		50
	Number of hidden layers		4
	Number of hidden neurons per layer		10
Convolutional Neural Network (CNN)	**All Layers Setting**	**Kernel initializer**	Glorot uniform Normalizer
		Bias initializer	Zeros
	Input Layer		6,6,1 layer
	Keras 2D Convolution Layer		32 Filters
			RELU activation function
			Kernel size: 1x1
			Strides: 1x1
			Dilation rate: 1x1
	Keras Max Pooling 2D Layer		Pool size: 2x2
			Strides: 2x2
	Keras 2D Convolution Layer		64 Filters
			RELU activation function
			Kernel size: 3x3
			Strides: 1x1
			Dilation rate: 1x1
	Keras Max Pooling 2D Layer		Pool size: 1x1
			Strides: 1x1
	Keras Flatten Layer		

	Keras Dense Layer		RELU activation function
			100 units
	Keras Dropout Layer		Drop rate: 0.5
	Keras Dense Layer (Output Layer)		Softmax Activation Function
			2 Units
Feedforward Neural Network (FNN)	**36-18-9-6-2 General Shape of Network**		
	All Layers Setting	**Kernel initializer**	Glorot uniform Normalizer
		Bias initializer	Zeros
	Input Layer		36 Shape
	Dense Layer		18 Units
			Tanh Activation Function
	Dense Layer		9 Units
			Tanh Activation Function
	Dropout Layer		Drop rate: 0.5
	Dense Layer		2 Units
			Tanh Activation Function
	Dense Layer (Output Layer)		2 Units
			Softmax Activation Function

CURRICULUM VITAE

April 2024

MEHMET EMİN ORHAN

EDUCATION

Master of Science in Bioengineering
Abdullah Gul University, Kayseri, Turkey (27.09.2021–Present)
Thesis Title: "In silico analysis of RNA interactions"
Advisor: Doç.Dr. Müşerref Duygu SAÇAR DEMİRCİ

Bachelor of Science in Molecular Biology and Genetics
Yıldız Technical University, Istanbul, Turkey (11.08.2017– 12.08.2021)

PROJECTS

- **NeRNA (2021-2023)**
- **CiliaMiner (2022-2023)**

PUBLICATION:

2024 - Orhan, M. E. et al., (2024) 'Bioinformatics Tools to Study the Role of miRNAs', *Human Health and Diseases*, Springer **(Under Review)**

2023 - Turan, M.G.*, **Orhan, M.E**.*, Cevik, S. et al. CiliaMiner: an integrated database for ciliopathy genes and ciliopathies. **Database** (2023) Vol. 2023: article ID baad047; DOI: https://doi.org/10.1093/database/baad047 (* indicates co-authors.)

2023 - Orhan, M. E. et al., (2023) NeRNA: a negative data generation framework for machine learning applications of non-coding RNAs. **Computers in Biology and Medicine**, 2023, 106861, ISSN 0010-4825, https://doi.org/10.1016/j.compbiomed.2023.106861

WEBSITE:

CiliaMiner - https://kaplanlab.shinyapps.io/ciliaminer/
(Shiny R package, CSS and JavaScript is used for generating database)

CODE ABILITY (https://github.com/Mehmeteminorhan)

R Language Programming
- Data Analysis, Data Visualization and Data Process
- Web development (Shiny, JavaScript and CSS)
- RNA-Seq and Single Cell Analysis
- Machine Learning and Deep Learning Development
- Pipeline Development

SQL
- Database Management

Python
- Biological Programming
- Workflow Development

KNIME
- Machine Learning and Deep Learning Development
- Workflow Development
- Curation Multiple Languages
- Data Analysis

Git
- Version Management
- GitHub Management

Linux
- NGS Analysis
- RNA-seq Analysis
- Small RNA-seq Analysis

COURSES AND CERTIFICATES

- Patika.dev - FMSS Bilişim Business Analyst Practicum
- IIENSTITU - Advanced MS Excel
- Global AI Hub - Introduction to AI, Robotics and Data
- BTK Academia - Introduction to Python Programming
- Coursera (University of Michigan) - Introduction to Python

AWARDS

2023 TUBITAK 2210-C National MSc Scholarship Program