PREDICTION OF miRNA SEQUENCE USING CNN AND LSTM

A PROJECT REPORT

Submitted by

ABINUS MERCY A (2019103501)

KANISHKAA R (2019103534)

KAVISHREE S (2019103537)

CS6611 – CREATIVE AND INNOVATIVE PROJECT

ANNA UNIVERSITY :: CHENNAI

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report **“ PREDICITON OF miRNA SEQUENCE USING CNN AND LSTM ”** is the bonafide work of **“ ABINUS MERCY A , KANISHKAA R, KAVISHREE S** ” who carried out the project work as a part of Creative and Innovative Project Laboratory.

DATE:	**SIGNATURE**
PLACE:	**Dr. K THANGARAMYA**
	COURSE IN-CHARGE
	TEACHING FELLOW
	DEPARTMENT OF CSE
	ANNA UNIVERSITY
	CHENNAI

TABLE OF CONTENTS

ABSTRACT

MicroRNAs(miRNA) are a class of endogenous small noncoding RNA molecule with a length of 22 nucleotides which plays an important role in the degradation and inhibition of mRNAs. MiRNA is a kind of essential drug targets for cancer therapy. MicroRNAs (miRNAs) are involved in a diverse variety of biological processes through regulating the expression of target genes in the post-transcriptional level. So, it is of great importance to discover the targets of miRNAs in biological research. But, due to the short length of miRNAs and limited sequence complementarity to their gene targets in animals, it is challenging to develop algorithms to predict the targets of miRNA accurately. Thus, it is important to provide an exhaustive analysis of the key miRNAs and the miRNA-mRNA interactions before applying the miRNA based therapeutics to the clinical trials. CNN and LSTM have proven their ability in feature extraction and natural language processing, respectively. So, the proposed model use their ability to process the language of RNAs, i.e., predicting sequence of microRNAs using the sequence of mRNA. The idea is to extract the features from sequence of mRNA using CNN and use LSTM network for prediction of miRNA. The model has learned the basic features such as seed match at first 2–8 nucleotides starting at the 50 end and counting toward the 30 end. Also, it was able to predict G-U wobble base pair in seed region. While validating on experimentally validated data, the model was able to predict on average 85 percent of miRNAs for specific mRNA and shows highest positive expression fold change of predicted targets on a microarray data generated using anti 25 miRNAs compare to other predicted tools. The developed model also predicted some novel miRNAs which are not yet annotated

CHAPTER 1

INTRODUCTION

Deep neural networks are machine learning algorithms which are inspired by biological neural networks, i.e., how our brain learns. Since their development in mid of 20th century, they did not get much application as they are computationally intensive. The inception of modern hardware systems, especially the GPUs (Graphical Processing Units) with more excellent computational capability, has allowed neural networks to regain popularity and applications. Convolutional Neural Networks(CNNs) and Recurrent Neural Networks(RNNs) with Long Short-Term Memory cells(LSTM) find their applications in various fields like image processing, speech recognition and natural language processing. ANNs, CNNs and LSTM have also been used in solving various biological problems including prediction of protein secondary structure, protein sub-cellular localisation, peptide binding to MHC-II molecules, prediction of methyladenosine sites in mRNA, image recognition of skin disorders etc.

Micro(mi)RNAs are small noncoding RNAs that regulate expression of the majority of the genes in the genome at either the messenger RNA (mRNA) level (by degrading mRNA) or the protein level (by blocking translation). miRNAs are thought to be components of vast regulatory networks. microRNA post-transcriptionally regulate gene expression by base-pairing to mRNAs. The human genome encodes for over 2200 microRNAs, which are mostly 28bp long, non-coding RNA molecules. Since one microRNA can target multiple gene transcripts, microRNAs are known to be involves in regulating gene expression and mRNA translation mechanisms.

Considering the massive presence of these regulatory RNAs, their diverse expression along with significant number of mRNA targets, it is not surprising that miRNAs have been playing crucial role in a broad range of diseases including immunological diseases, cancers, and various skin diseases. The connection between miRNA and mRNA can be resolved experimentally utilising different techniques, for example, HITS-CLIP, PAR-CLIP, CLASH etc.

The primary purpose of the work is to utilise the speciality of CNN and LSTM architecture in prediction of miRNA sequences based on target mRNA segment where they bound. The problem stated is a typical sequence to sequence prediction problem where

LSTMs are most suitable. Currently, the field is focused primarily on identifying novel targets of individual miRNAs.

1.1 PROBLEM STATEMENT

The action of miRNAs on their mRNA targets is difficult to characterize, because each miRNA has multiple mRNA targets and vice versa; therefore, the correct identification of an interaction remains a challenge. The objective of the project is to predict and validate the miRNA:mRNA interactions using CNN and LSTM.

1.2 PROBLEM DESCRIPTION

Prediction of microRNA targets, which has been a mind storming challenge for scientific community for decades, is a basic fundamental step in finding microRNA-mRNA target association. The current methods for microRNA target predictions incorporate various computational methodologies, from the demonstration of physical association algorithms to the application of machine learning algorithms. Most of these algorithms uses artificially negative set data for training, as experimentally validated negative set is very tedious to found, so these algorithms show low sensitivity in real data.

Because of not fully understood rules that govern miRNAs targeting process and different training datasets for different algorithms, there is limited overlap between the targets that are predicted by various programs. So, it is still a challenge to develop more reliable computation methods based on more accurate miRNA. Having a good progress in CNN and LSTM, the model uses this technology to predict microRNA targets.

1.3 OBJECTIVE

The objective of the model is to predict the target gene of miRNAs through scanning the full length of gene transcripts. The model can also predict some novel miRNAs which are not yet annotated. Sequence level interaction is basic criteria for mRNA and miRNA interaction, so the crust of these interactions can be found in sequence level features which are hard to generate. So, we propose Seq2Seq architecture which is proven to use in sequence level NLP data. The model needs to identify the accessible regions in a mRNA where miRNAs could bind and then use the mRNA segments of those regions for prediction miRNAs which can target mRNA.

CHAPTER 2

LITERATURE SURVEY

Xueming Zheng et al,2018 [8] proposed Multilayer CNN which automatically extract pattern from canonical and non-canonical pairing between the miRNAs and its targets. But the complicated molecular interaction networks in the cell affects the interactions of miRNAs and target mRNAs.

Jianrong Yan et al,2020 [3] developed SDA (Stacked Denoising Autoencoder) and CAE (Convolutional Denoising Autoencoder) which are used to pre-train in first step by original sequence and structured data respectively. The encoders of the two models are fused for prediction model. The added upstream and downstream sequence information improves the prediction accuracy of the model. The two-dimensional convolution is more effective for feature extraction of structure data. High sensitivity, since autoencoders can be more sensitive to input errors than manual approaches. Misunderstanding important variables, Imperfect decoding, Training the wrong use case.

Nafiesh Sedaghat et al,2018 [6] proposed three versions of each SVM classifier which were used to identify miRNA-mRNA interactions by considering only binding features, only network features and both set of features. Binary SVM, the training data comprised both positive and negative examples. It doesn’t perform well when large data set is used because required training time is higher.

Ghosal et al,2018 [1] developed Support Vector Machine(SVM) based model for miRNA target prediction using recent CLIP-seq data which solves higher computational burden. More productive in high dimensional places. This classifier is not suitable for larger datasets. It doesn’t perform well when large data set is used because required training time is higher.

Vinani A.Kashara & Maria C, 2018 [2] developed multiclass CoNN algorithm which achieves results with the advantage of low computational cost and no model selection. Maybe be leads to overfitting.

Yiqun Xiao et al,2018 [9] proposed a sequence-to-sequence model to encode MiRNA sequences and predict their subcellular locations. The model learns high level features hidden in the MiRNA sequences. Encoder helps in mapping the input sequence into hidden states that lie in a low dimensional vector space. Decoder helps to determine the output of the next location.

Mohammed Q.Shatwani,2016 [5] proposed Adaptive Boosting(AdaBoost) algorithm to create the ensemble classifiers that consist of several SVM classifiers which helps in better prediction. Decreases false predictions using classifiers. This methodology does incur a cost as it is computationally heavy, with each other classifier have to be executed before the final result.

Karol Szafranski et al,2015 [4] developed SVM model for better accuracy and faster prediction. SVM is not suitable for large datasets. It will not perform properly when the number of features for each datapoint exceeds the number of training data samples.

Shuang Cheng et al,2015 [7] proposed constraint relaxation method to construct balanced datasets and training the classifier using CNN. Multiple features can be extracted at each location using CNN. CNN avoids the effect of incorrect domain model on prediction. The algorithm’s predictions have overlaps because the features they are derived from similar characteristics that consist of conservation, complementation and accessibility.

CHAPTER 3

PROPOSED SYSTEM

3.1 SYSTEM ARCHITECTURE

In the present work, we have used CNN to extract sequence features from input mRNA segment and fed these features to a LSTM system stacked in a sequence to sequence architecture. This architecture generally consists of two LSTMs known as encoder and decoder. The extracted features from CNN are forwarded to encoder LSTM, one by one in time steps to obtain a fixed dimensional vector of internal states; these internal states are used as initial states by decoder LSTM to extract the output sequence according to its initial state vector. This LSTM is a RNN language model but its training is based on the input sequence vector, wherein the encoder processes the features from targeted mRNA sequences and returns its hidden internal states, which are further used by decoder LSTM as condition or context for prediction of miRNA sequences.

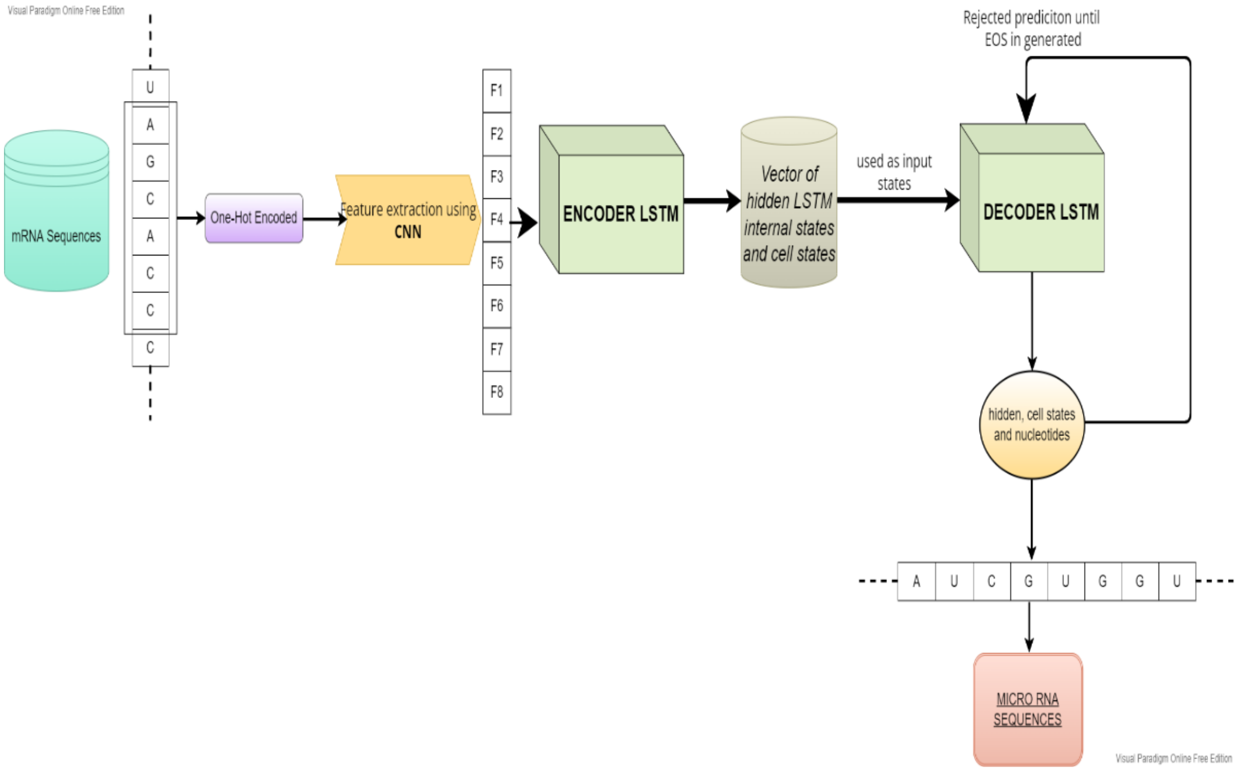

Visual Paradigm Online Free Edition
mRNA Sequences
U
A
G
C
A
C
C
C
One-Hot Encoded
Feature extraction using CNN
F1
F2
F3
F4
F5
F6
F7
F8
ENCODER LSTM
Vector of hidden LSTM internal states and cell states
used as input states
DECODER LSTM
Rejected prediciton until EOS in generated
hidden, cell states and nucleotides
A
U
C
G
U
G
G
U
MICRO RNA SEQUENCES
Visual Paradigm Online Free Edition

Fig 3.1. System Architecture diagram

3.2 PROPOSED METHODOLOGY

Table 3.1. Module Design

MODULE	INPUT	OUTPUT
Data Curation	microRNA name	Dataset with its target binding sites
Obtaining Surface Area Accessibility Regions	Gene symbol	Transcript IDs and mRNA segments
Data preprocessing	Dataset from web crawler	Encoded nucleotide with default length
Neural Network & LSTM Encoder	microRNA and mRNA sequences	Extracted features from the sequences with Hidden internal & cell states of every cell
LSTM decoder	Output of the Encoder along with features extracted using CNN	Predicted miRNA sequence
Filtering and mapping	Predicted miRNAs and their corresponding mRNAs	Predicted microRNA IDs

3.3MODULES DESCRIPTION

3.3.1 DATA CURATION

For seq2seq architecture, sequence of microRNAs and their corresponding binding target sites in mRNA are required.microRNA sequence were retrieved from miRBase release 22,March 2020.An online web crawler is used to store the retrieved data in CSV file. The CSV file contains microRNA Name, Gene Symbol, Chromosome location of the target binding site according to Ensemble Human (GRCh38. p12) annotation(25350 datapoints).

3.3.2 OBTAINING SURFACE AREA ACCESSIBILITY REGIONS

Sequences of all protein-coding transcripts of a query gene and their 3'UTRs were retrieved using Ensembl REST API. As Ensembl REST API accepts transcript IDs as input, so a local SQLite database was developed for mapping Gene Symbols with all respective protein-coding Ensembl transcript IDs.We used RNAplfold program from ViennaRNA Package 2.0 to compute locally stable secondary structure − pair probabilities. RNAplfold select accessible 8-mers in 3'UTR region for finding the microRNA binding site in respective mRNA, i.e., 24-mer, 26-mer and 28-mer which are in polarity 3′ to 5′. Then these mRNA segments will be used for predicting respective microRNA sequences by our trained model.

3.3.3 DATA PREPROCESSING AND PREPARATION

In the dataset we analyzed that the difference between the lengths of pairs of microRNA and mRNA varied from 0 to 18. To bring a uniformity in different in length of sequence so that our model may extract features based on patterns in the sequence not from the length of sequences, we looked into the distribution of the pair sequence length difference. We found that the distribution was skewed, having around 90 percent of the difference in length pair of sequences between 0 to 6. So a threshold of 6 was set, and the microRNA and mRNA sequences having sequence length difference of 6 or less were used for further analysis.Categorical features such as mRNA and microRNA nucleotides first need to be encoded numerically. They are typically represented as binary vectors with all but one entry set to zero, which indicates the category (one-hot coded embedding). One-hot coded strings which are input of a deep neural network model should be of same length, i.e., all microRNAs should be of same length, and all mRNA-target site sequence should be of same length, which is not the scenario of real world. So, we took the largest sequence among the microRNA sequences as a default length of all microRNAs which turned out to be 28. We made other short length microRNAs of same length by padding them with zeros. In the same way, we treated the mRNA sequences, and their default length turned out to be 29.

3.3.4 NEURAL NETWORK AND LSTM ENCODER

Window size of 8 for extracting 128 features. Then we used a dense layer of 128 neurons for adjusting weight of 128 features which were extracted from mRNA and miRNA sequences, which were then fed into LSTM networks. LSTMs are stacked in sequence-to-sequence architecture were two LSTMs are used, one is encoder which processes the input sequence i.e., mRNA sequence features. Encoder LSTMs are programed for returning the hidden internal and cell states of every cell of that layer.These outputs along with the feature extracted from CNN on miRNA sequences were used to initialise the state of the decoder. Therefore every time the encoder model processes an input sequence, the last internal states of the encoder model are used as the starting point for prediction of the first character of the miRNA sequence. Encoder is further connected with a softmax gives the probability of nucleotide at that step(on-hot encoded) and compare with target data where error is calculated using categorical cross entropy loss function.

3.3.5 LSTM DECODER

The decoder LSTM needs the hidden, cell states and initial ′t′ token from the encoder LSTM as its new initial state.The decoder will be called recursively for each nucleotide that is to be generated for prediction of miRNA sequence.On the first call, the hidden, cell states and ′t′ from the encoder will be used to initialize the decoder LSTM layer.From second onward recursive calls to the decoder, the last hidden, cell states and nucleotide should be provided from the its output of the previous steps till ′n′ is predicted.The decoder outputs the hidden and cell states along with the predicted character on each call, so that these states can be assigned to a variable and used on each subsequent recursive call for a given CNN output of sequence features of mRNA for prediction of miRNA.

3.3.6 FILTERING AND MAPPING

Predicted miRNAs and their corresponding mRNAs are filtered by a threshold of ΔG of -9 (the average ΔG of our training set).Then these predicted microRNA sequences will be mapped to their microRNA IDs using BLASTx algorithm on local database containing microRNA ID and respective sequences retrieved from mirBase Release 22, March 2018- hence giving a list of predicted microRNA IDs as output.The model can also predict some novel miRNAs which are not yet annotated.

CHAPTER 4

RESULT AND DISCUSSION

4.1 EVALUATION METRICS

Using the trained model, we will calculate the classifier performance on the test dataset in terms of Sensitivity, Specificity, F1-Score, and accuracy.

SENSITIVITY

$$\text{Sen.} = \frac{TP}{TP+FN} \qquad \text{eqn(1)}$$

SPECIFICITY

$$\text{Spe.} = \frac{TN}{TN+FP} \qquad \text{eqn(2)}$$

F1-SCORE

$$\text{F1} = \frac{2*TP}{2*TP+FP+FN} \qquad \text{eqn(3)}$$

ACCURACY

$$\text{Acc.}=\frac{TP+TN}{TP+TN+FP+FN} \qquad \text{eqn(4)}$$

TP - *TRUE POSITIVE*

TN - *TRUE NEGATIVE*

FP - *FALSE POSITIVE*

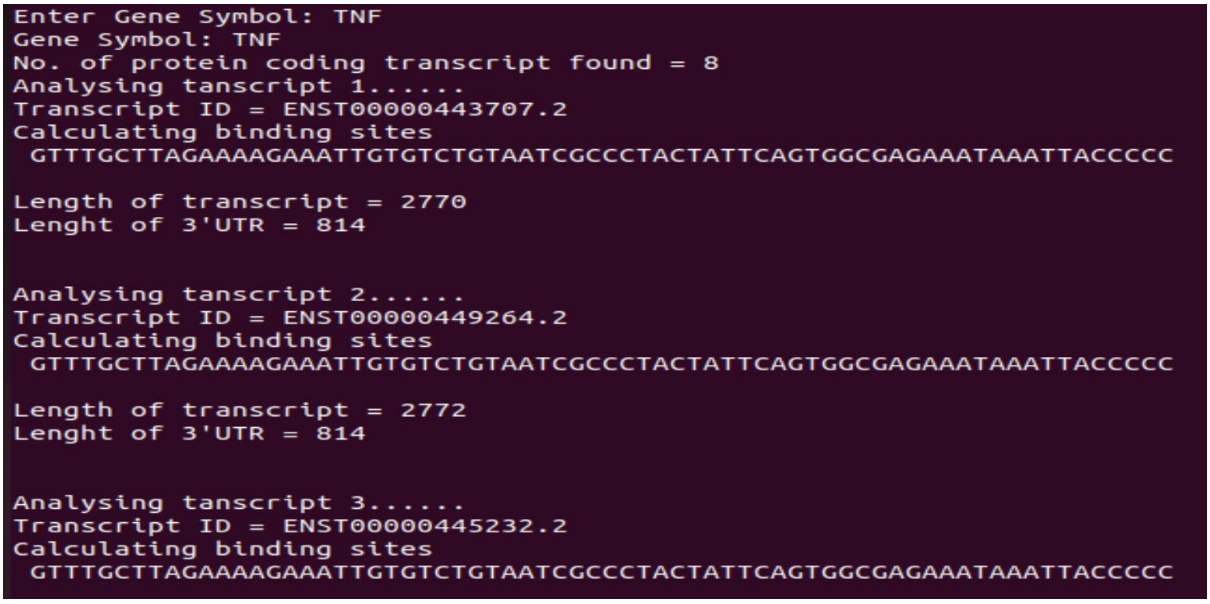
FN - *FALSE NEGATIVE*

4.2 RESULTS OBTAINED

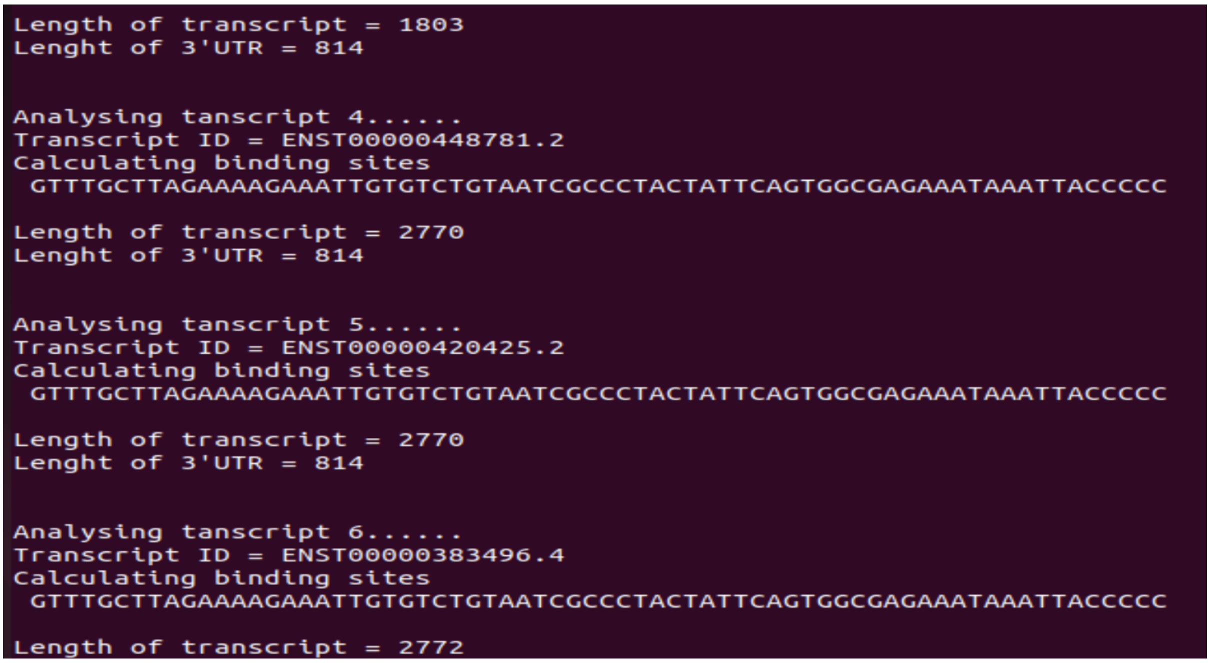
Table 4.1 Results

TRAINING ACCURACY	85.7986
VALIDATION ACCURACY	86.234
TRAINING LOSS	0.128
VALIDATION LOSS	0.087
F1 - SCORE	81.333

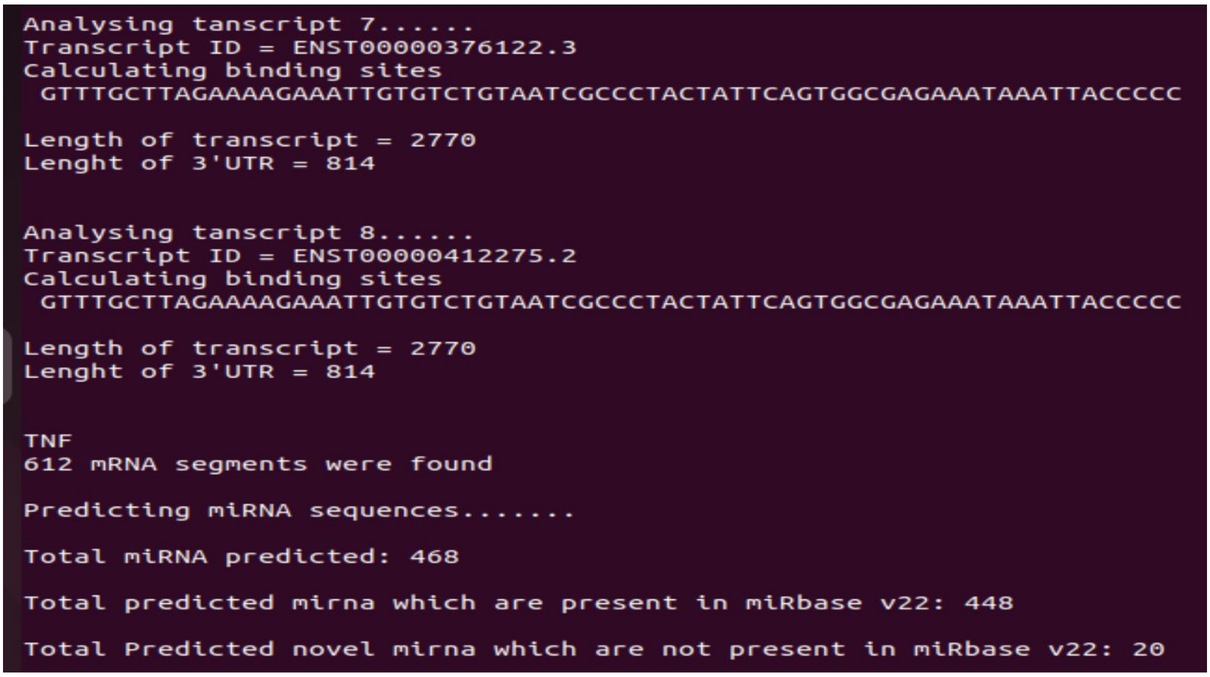
While running the program, the user will be prompted to enter a Gene symbol. Then the Gene symbol will be used to retrieve all associated protein-coding Ensembl transcript IDs. RNA sequence of these transcript IDs and location of 3’UTR will be retrieved using Ensembl REST API. These sequences will be forwarded to RNAplfold of ViennaRNA package for finding accessibility region or accessible 8-mers. Subsequently, RNAplfold select accessible 8- mers in 3’UTR region for finding the microRNA binding site in respective mRNA, i.e., 24-mer, 26-mer and 28-mer which are in polarity 30 to 50 .



```
Enter Gene Symbol: TNF
Gene Symbol: TNF
No. of protein coding transcript found = 8
Analysing tanscript 1......
Transcript ID = ENST00000443707.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2770
Lenght of 3'UTR = 814


Analysing tanscript 2......
Transcript ID = ENST00000449264.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2772
Lenght of 3'UTR = 814


Analysing tanscript 3......
Transcript ID = ENST00000445232.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC
```

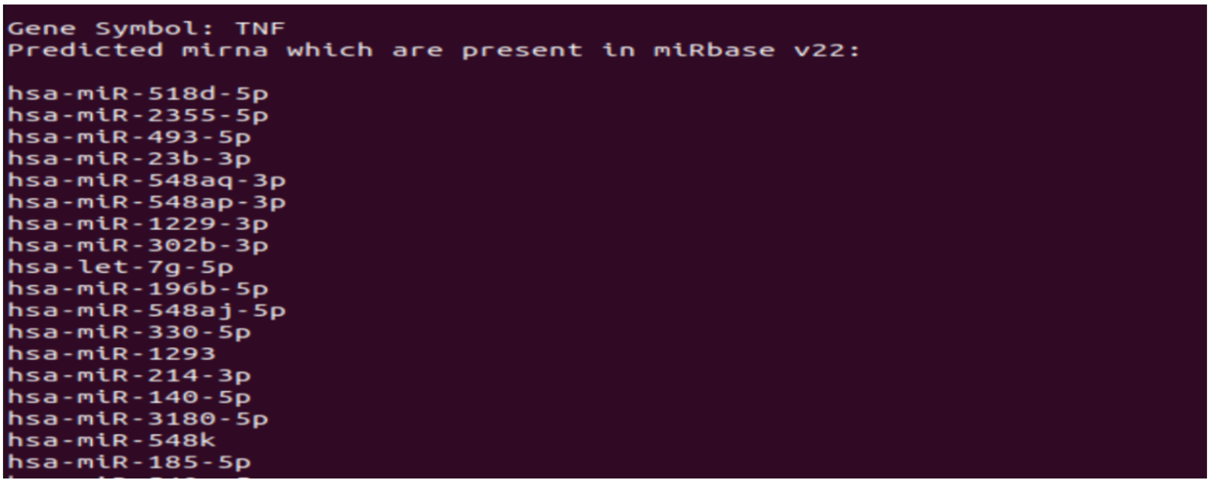


```
Length of transcript = 1803
Lenght of 3'UTR = 814


Analysing tanscript 4......
Transcript ID = ENST00000448781.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2770
Lenght of 3'UTR = 814


Analysing tanscript 5......
Transcript ID = ENST00000420425.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2770
Lenght of 3'UTR = 814


Analysing tanscript 6......
Transcript ID = ENST00000383496.4
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2772
```

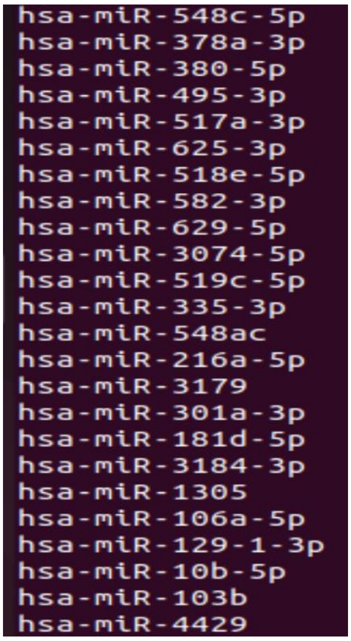


```
Analysing tanscript 7......
Transcript ID = ENST00000376122.3
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2770
Lenght of 3'UTR = 814


Analysing tanscript 8......
Transcript ID = ENST00000412275.2
Calculating binding sites
 GTTTGCTTAGAAAAGAAATTGTGTCTGTAATCGCCCTACTATTCAGTGGCGAGAAATAAATTACCCCC

Length of transcript = 2770
Lenght of 3'UTR = 814


TNF
612 mRNA segments were found

Predicting miRNA sequences.......

Total miRNA predicted: 468

Total predicted mirna which are present in miRbase v22: 448

Total Predicted novel mirna which are not present in miRbase v22: 20
```

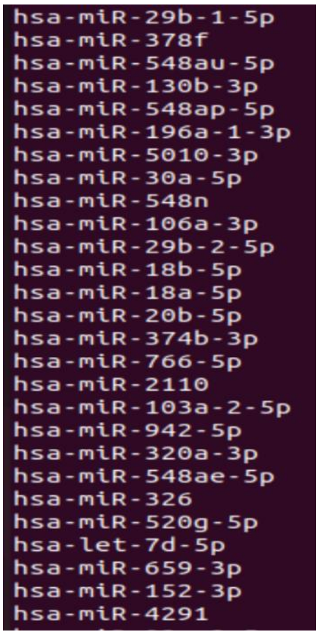
Then these mRNA segments will be used for predicting respective microRNA sequences by our trained model. Predicted miRNAs and their corresponding mRNAs are filtered by a threshold of DG of 9 (the average DG of our training set). Then these predicted microRNA sequences will be mapped to their microRNA IDs using BLASTx algorithm on local database containing microRNA ID and respective sequences retrieved from mirBase Release 22, March 2018- hence giving a list of predicted microRNA IDs as output.



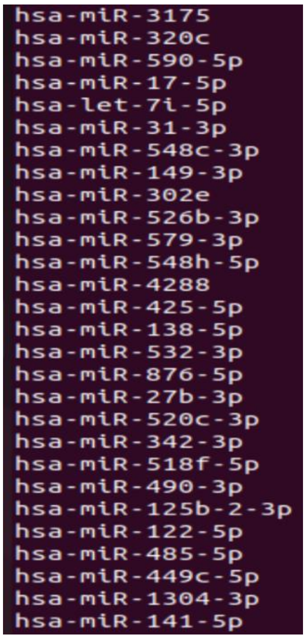
```
Gene Symbol: TNF
Predicted mirna which are present in miRbase v22:

hsa-miR-518d-5p
hsa-miR-2355-5p
hsa-miR-493-5p
hsa-miR-23b-3p
hsa-miR-548aq-3p
hsa-miR-548ap-3p
hsa-miR-1229-3p
hsa-miR-302b-3p
hsa-let-7g-5p
hsa-miR-196b-5p
hsa-miR-548aj-5p
hsa-miR-330-5p
hsa-miR-1293
hsa-miR-214-3p
hsa-miR-140-5p
hsa-miR-3180-5p
hsa-miR-548k
hsa-miR-185-5p
```



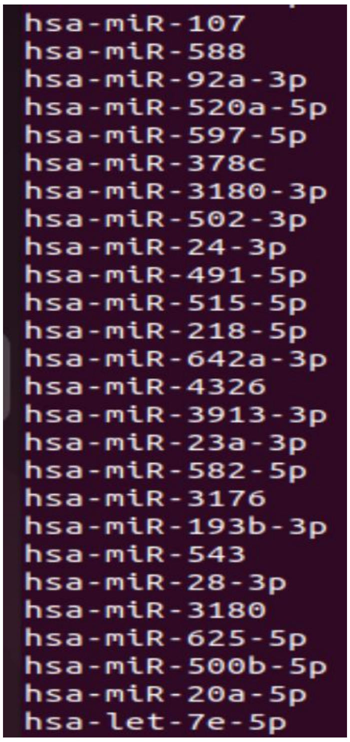
```
hsa-miR-548c-5p
hsa-miR-378a-3p
hsa-miR-380-5p
hsa-miR-495-3p
hsa-miR-517a-3p
hsa-miR-625-3p
hsa-miR-518e-5p
hsa-miR-582-3p
hsa-miR-629-5p
hsa-miR-3074-5p
hsa-miR-519c-5p
hsa-miR-335-3p
hsa-miR-548ac
hsa-miR-216a-5p
hsa-miR-3179
hsa-miR-301a-3p
hsa-miR-181d-5p
hsa-miR-3184-3p
hsa-miR-1305
hsa-miR-106a-5p
hsa-miR-129-1-3p
hsa-miR-10b-5p
hsa-miR-103b
hsa-miR-4429
```



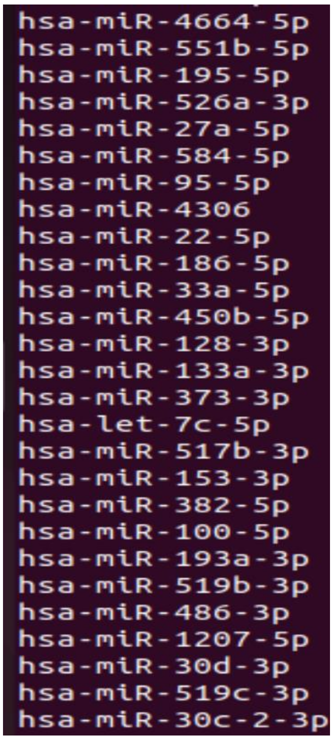
```
hsa-miR-29b-1-5p
hsa-miR-378f
hsa-miR-548au-5p
hsa-miR-130b-3p
hsa-miR-548ap-5p
hsa-miR-196a-1-3p
hsa-miR-5010-3p
hsa-miR-30a-5p
hsa-miR-548n
hsa-miR-106a-3p
hsa-miR-29b-2-5p
hsa-miR-18b-5p
hsa-miR-18a-5p
hsa-miR-20b-5p
hsa-miR-374b-3p
hsa-miR-766-5p
hsa-miR-2110
hsa-miR-103a-2-5p
hsa-miR-942-5p
hsa-miR-320a-3p
hsa-miR-548ae-5p
hsa-miR-326
hsa-miR-520g-5p
hsa-let-7d-5p
hsa-miR-659-3p
hsa-miR-152-3p
hsa-miR-4291
```



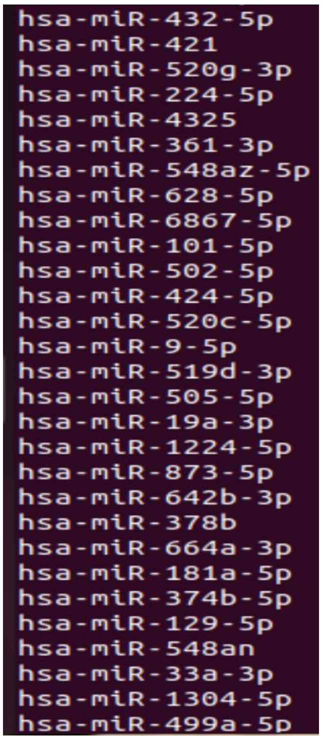
```
hsa-miR-3175
hsa-miR-320c
hsa-miR-590-5p
hsa-miR-17-5p
hsa-let-7i-5p
hsa-miR-31-3p
hsa-miR-548c-3p
hsa-miR-149-3p
hsa-miR-302e
hsa-miR-526b-3p
hsa-miR-579-3p
hsa-miR-548h-5p
hsa-miR-4288
hsa-miR-425-5p
hsa-miR-138-5p
hsa-miR-532-3p
hsa-miR-876-5p
hsa-miR-27b-3p
hsa-miR-520c-3p
hsa-miR-342-3p
hsa-miR-518f-5p
hsa-miR-490-3p
hsa-miR-125b-2-3p
hsa-miR-122-5p
hsa-miR-485-5p
hsa-miR-449c-5p
hsa-miR-1304-3p
hsa-miR-141-5p
```



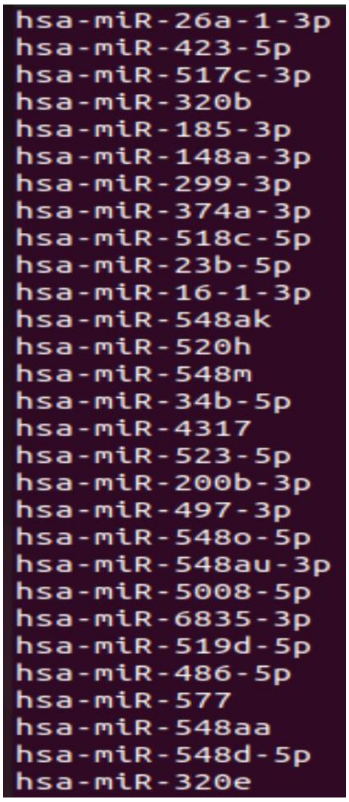
```
hsa-miR-107
hsa-miR-588
hsa-miR-92a-3p
hsa-miR-520a-5p
hsa-miR-597-5p
hsa-miR-378c
hsa-miR-3180-3p
hsa-miR-502-3p
hsa-miR-24-3p
hsa-miR-491-5p
hsa-miR-515-5p
hsa-miR-218-5p
hsa-miR-642a-3p
hsa-miR-4326
hsa-miR-3913-3p
hsa-miR-23a-3p
hsa-miR-582-5p
hsa-miR-3176
hsa-miR-193b-3p
hsa-miR-543
hsa-miR-28-3p
hsa-miR-3180
hsa-miR-625-5p
hsa-miR-500b-5p
hsa-miR-20a-5p
hsa-let-7e-5p
```



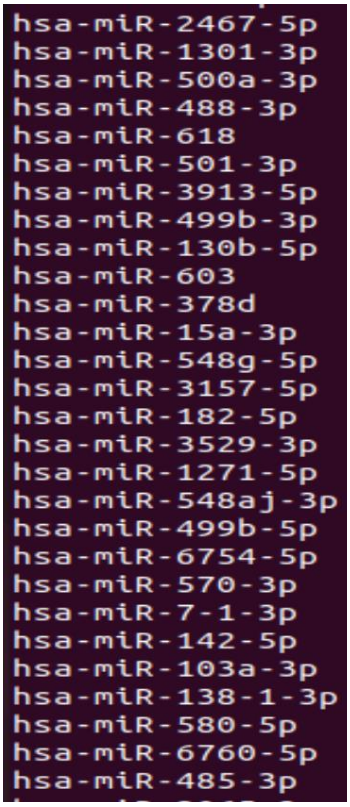
```
hsa-miR-4664-5p
hsa-miR-551b-5p
hsa-miR-195-5p
hsa-miR-526a-3p
hsa-miR-27a-5p
hsa-miR-584-5p
hsa-miR-95-5p
hsa-miR-4306
hsa-miR-22-5p
hsa-miR-186-5p
hsa-miR-33a-5p
hsa-miR-450b-5p
hsa-miR-128-3p
hsa-miR-133a-3p
hsa-miR-373-3p
hsa-let-7c-5p
hsa-miR-517b-3p
hsa-miR-153-3p
hsa-miR-382-5p
hsa-miR-100-5p
hsa-miR-193a-3p
hsa-miR-519b-3p
hsa-miR-486-3p
hsa-miR-1207-5p
hsa-miR-30d-3p
hsa-miR-519c-3p
hsa-miR-30c-2-3p
```



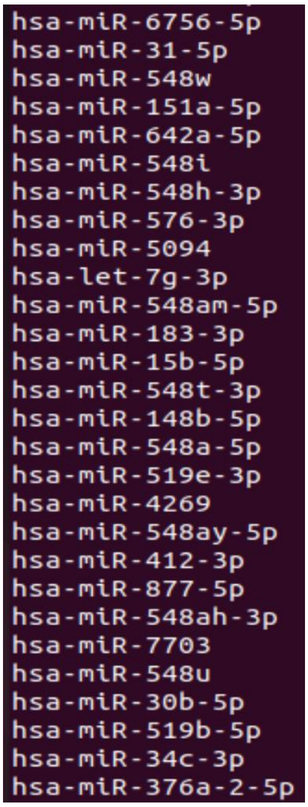
```
hsa-miR-432-5p
hsa-miR-421
hsa-miR-520g-3p
hsa-miR-224-5p
hsa-miR-4325
hsa-miR-361-3p
hsa-miR-548az-5p
hsa-miR-628-5p
hsa-miR-6867-5p
hsa-miR-101-5p
hsa-miR-502-5p
hsa-miR-424-5p
hsa-miR-520c-5p
hsa-miR-9-5p
hsa-miR-519d-3p
hsa-miR-505-5p
hsa-miR-19a-3p
hsa-miR-1224-5p
hsa-miR-873-5p
hsa-miR-642b-3p
hsa-miR-378b
hsa-miR-664a-3p
hsa-miR-181a-5p
hsa-miR-374b-5p
hsa-miR-129-5p
hsa-miR-548an
hsa-miR-33a-3p
hsa-miR-1304-5p
hsa-miR-499a-5p
```



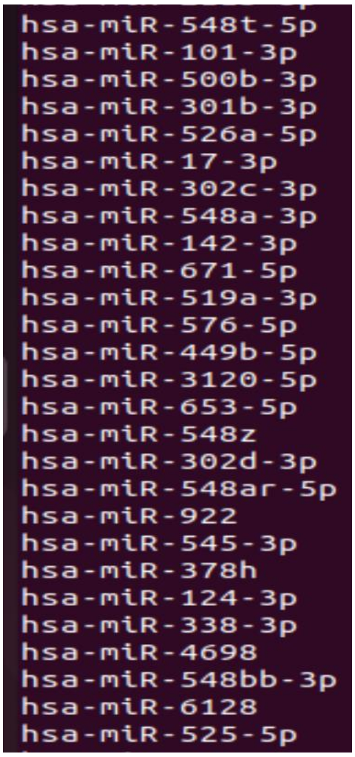
```
hsa-miR-26a-1-3p
hsa-miR-423-5p
hsa-miR-517c-3p
hsa-miR-320b
hsa-miR-185-3p
hsa-miR-148a-3p
hsa-miR-299-3p
hsa-miR-374a-3p
hsa-miR-518c-5p
hsa-miR-23b-5p
hsa-miR-16-1-3p
hsa-miR-548ak
hsa-miR-520h
hsa-miR-548m
hsa-miR-34b-5p
hsa-miR-4317
hsa-miR-523-5p
hsa-miR-200b-3p
hsa-miR-497-3p
hsa-miR-548o-5p
hsa-miR-548au-3p
hsa-miR-5008-5p
hsa-miR-6835-3p
hsa-miR-519d-5p
hsa-miR-486-5p
hsa-miR-577
hsa-miR-548aa
hsa-miR-548d-5p
hsa-miR-320e
```



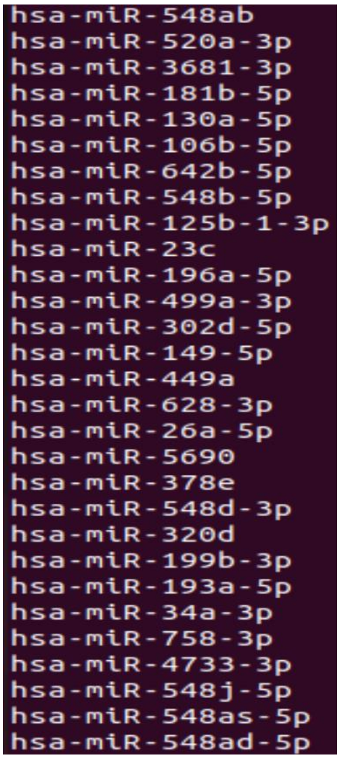
```
hsa-miR-2467-5p
hsa-miR-1301-3p
hsa-miR-500a-3p
hsa-miR-488-3p
hsa-miR-618
hsa-miR-501-3p
hsa-miR-3913-5p
hsa-miR-499b-3p
hsa-miR-130b-5p
hsa-miR-603
hsa-miR-378d
hsa-miR-15a-3p
hsa-miR-548g-5p
hsa-miR-3157-5p
hsa-miR-182-5p
hsa-miR-3529-3p
hsa-miR-1271-5p
hsa-miR-548aj-3p
hsa-miR-499b-5p
hsa-miR-6754-5p
hsa-miR-570-3p
hsa-miR-7-1-3p
hsa-miR-142-5p
hsa-miR-103a-3p
hsa-miR-138-1-3p
hsa-miR-580-5p
hsa-miR-6760-5p
hsa-miR-485-3p
```



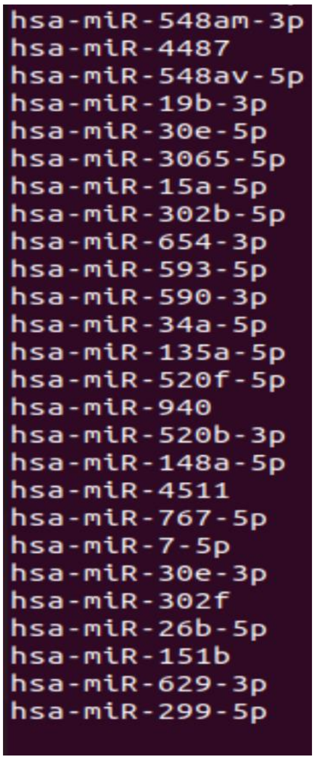
```
hsa-miR-6756-5p
hsa-miR-31-5p
hsa-miR-548w
hsa-miR-151a-5p
hsa-miR-642a-5p
hsa-miR-548i
hsa-miR-548h-3p
hsa-miR-576-3p
hsa-miR-5094
hsa-let-7g-3p
hsa-miR-548am-5p
hsa-miR-183-3p
hsa-miR-15b-5p
hsa-miR-548t-3p
hsa-miR-148b-5p
hsa-miR-548a-5p
hsa-miR-519e-3p
hsa-miR-4269
hsa-miR-548ay-5p
hsa-miR-412-3p
hsa-miR-877-5p
hsa-miR-548ah-3p
hsa-miR-7703
hsa-miR-548u
hsa-miR-30b-5p
hsa-miR-519b-5p
hsa-miR-34c-3p
hsa-miR-376a-2-5p
```



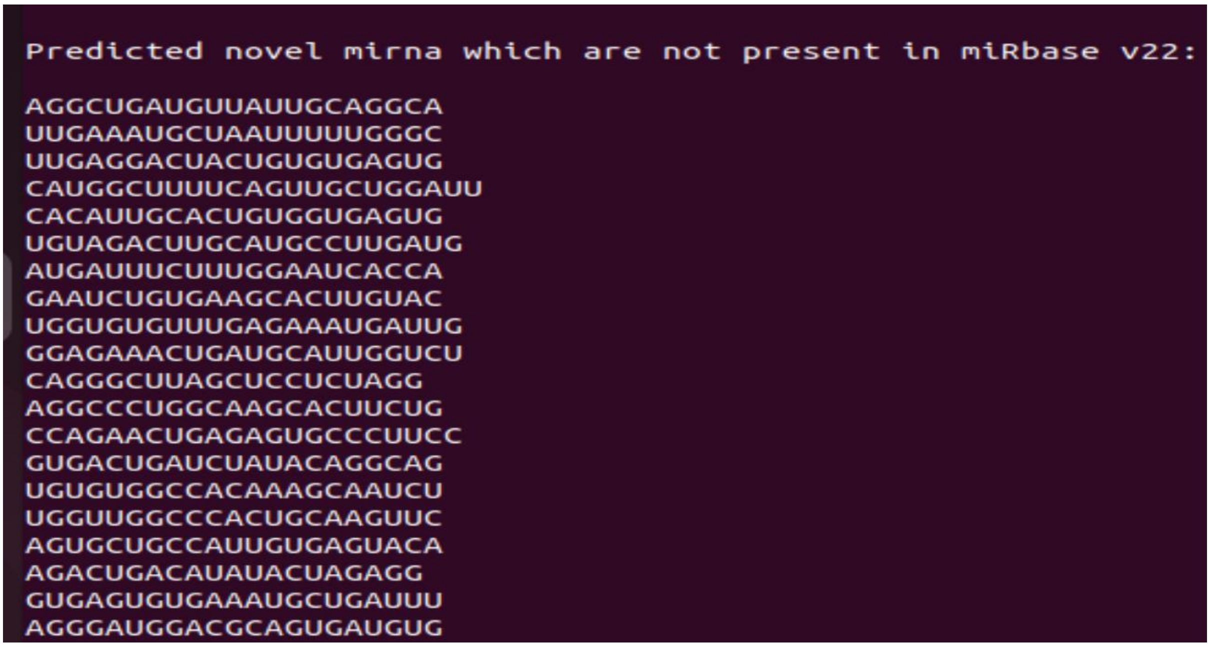
```
hsa-miR-548t-5p
hsa-miR-101-3p
hsa-miR-500b-3p
hsa-miR-301b-3p
hsa-miR-526a-5p
hsa-miR-17-3p
hsa-miR-302c-3p
hsa-miR-548a-3p
hsa-miR-142-3p
hsa-miR-671-5p
hsa-miR-519a-3p
hsa-miR-576-5p
hsa-miR-449b-5p
hsa-miR-3120-5p
hsa-miR-653-5p
hsa-miR-548z
hsa-miR-302d-3p
hsa-miR-548ar-5p
hsa-miR-922
hsa-miR-545-3p
hsa-miR-378h
hsa-miR-124-3p
hsa-miR-338-3p
hsa-miR-4698
hsa-miR-548bb-3p
hsa-miR-6128
hsa-miR-525-5p
```



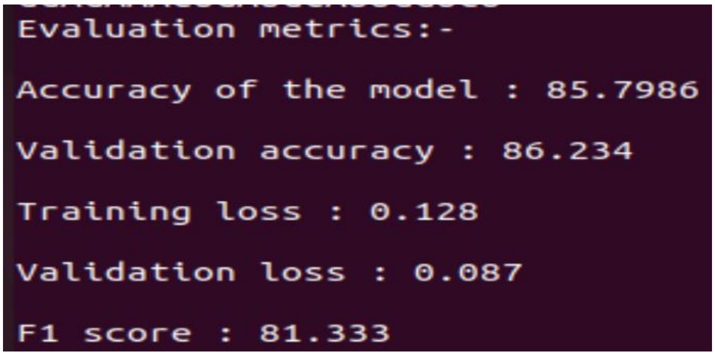
```
hsa-miR-548ab
hsa-miR-520a-3p
hsa-miR-3681-3p
hsa-miR-181b-5p
hsa-miR-130a-5p
hsa-miR-106b-5p
hsa-miR-642b-5p
hsa-miR-548b-5p
hsa-miR-125b-1-3p
hsa-miR-23c
hsa-miR-196a-5p
hsa-miR-499a-3p
hsa-miR-302d-5p
hsa-miR-149-5p
hsa-miR-449a
hsa-miR-628-3p
hsa-miR-26a-5p
hsa-miR-5690
hsa-miR-378e
hsa-miR-548d-3p
hsa-miR-320d
hsa-miR-199b-3p
hsa-miR-193a-5p
hsa-miR-34a-3p
hsa-miR-758-3p
hsa-miR-4733-3p
hsa-miR-548j-5p
hsa-miR-548as-5p
hsa-miR-548ad-5p
```



```
hsa-miR-548am-3p
hsa-miR-4487
hsa-miR-548av-5p
hsa-miR-19b-3p
hsa-miR-30e-5p
hsa-miR-3065-5p
hsa-miR-15a-5p
hsa-miR-302b-5p
hsa-miR-654-3p
hsa-miR-593-5p
hsa-miR-590-3p
hsa-miR-34a-5p
hsa-miR-135a-5p
hsa-miR-520f-5p
hsa-miR-940
hsa-miR-520b-3p
hsa-miR-148a-5p
hsa-miR-4511
hsa-miR-767-5p
hsa-miR-7-5p
hsa-miR-30e-3p
hsa-miR-302f
hsa-miR-26b-5p
hsa-miR-151b
hsa-miR-629-3p
hsa-miR-299-5p
```



```
Predicted novel mirna which are not present in miRbase v22:

AGGCUGAUGUUAUUGCAGGCA
UUGAAAUGCUAAUUUUUGGGC
UUGAGGACUACUGUGUGAGUG
CAUGGCUUUUCAGUUGCUGGAUU
CACAUUGCACUGUGGUGAGUG
UGUAGACUUGCAUGCCUUGAUG
AUGAUUUCUUUGGAAUCACCA
GAAUCUGUGAAGCACUUGUAC
UGGUGUGUUUGAGAAAUGAUUG
GGAGAAACUGAUGCAUUGGUCU
CAGGGCUUAGCUCCUCUAGG
AGGCCCUGGCAAGCACUUCUG
CCAGAACUGAGAGUGCCCUUCC
GUGACUGAUCUAUACAGGCAG
UGUGUGGCCACAAAGCAAUCU
UGGUUGGCCCACUGCAAGUUC
AGUGCUGCCAUUGUGAGUACA
AGACUGACAUAUACUAGAGG
GUGAGUGUGAAAUGCUGAUUU
AGGGAUGGACGCAGUGAUGUG
```



```
Evaluation metrics:-

Accuracy of the model : 85.7986

Validation accuracy : 86.234

Training loss : 0.128

Validation loss : 0.087

F1 score : 81.333
```

4.3 RESULT ANALYSIS

After training for 100 epochs, we found that with every epoch, our model began to understand the key features such as Watson and Crick base pairing in seed region i.e., it started predicting right complementary. Also, it could generate miRNA sequences having G:U wooble base pairs with given mRNA. At the end of training, the training accuracy of our model for predicting microRNA sequence based on its binding site in mRNA was around 85 percent, while in validation set, we obtained a 1 percent increase in accuracy, i.e., around 86 percent.

As training accuracy is less than validation accuracy, we can state that our model is not overfitted. Also, there was a decrease in validation loss than training loss, i.e., training loss was 0.128 while validation loss was 0.087, which corroborated the same. From training matrices, it could be seen that microRNA sequences can be predicted with up to 80 percent similarity using its target binding segment in mRNA.

CHAPTER 5

CONCLUSION

In the present work, we have used CNN and LSTM in a sequence to-sequence architecture to perform character level prediction of miRNAs using their targeted mRNA segment. Since identifying patterns and features for binding mRNA and its complimentary miRNA is a very tedious job at human level, we utilized CNN. RNNs with LSTM cells have proven applications in speech processing and natural language processing. Therefore, we included it to process the genetic language i.e., ‘A’; ‘U’; ‘G’; ‘C’ in case of RNAs. After confirming that the model has learned the basic features such as seed match at first 2–8 nucleotides starting at the 50 end and counting toward the 30 end and that it was able to predict G-U wobble base pair in seed region, we used this model for prediction of microRNA from mRNA sequence. The developed model also predicted some novel miRNAs which are not yet annotated.

5.1 FUTURE WORKS

While this project is only a first step towards predicting sequence of miRNAs, in future, we intend to use a robust deep learning model which could understand most of the features of miRNA and mRNA present in their sequence. Also, miRNAs have been proven clinically to be associated with various diseases; this approach can be useful for finding target mRNAs and act as an additional knowledge resource in prognosis of such conditions.

APPENDICES

IMPLEMENTATION

Run_model.py

```
list_mrna = get_seq_to_predict(GeneSymbol)

predicted_mirna=[]
for mrna in predict_list:
    micro= predict_mirna(mrna)[:-1]
    mrna=mrna[::-1]
    if RNA.fold(mrna[:]+'LLLLLLLL'+micro[:])[1]<-3:
        predicted_mirna.append(micro)
list_a=[]
for mrna,mirna in zip(predict_list,predicted_mirna):
    list_a.append([mrna,mirna])
predicted_mirna=list(set(predicted_mirna))
mirna_prediction=[]
pair = []
for mrna,mirna in list_a:
    mirna_name = Blast_seq(mirna)
    if mirna_name !=None:
        for name in mirna_name:
            mirna_prediction.append(name)
            pair.append([name,mirna,mrna])
    else:
        mirna_prediction.append(mirna)
```

Secondary_structure.py:

```
conn = sqlite3.connect('/data/linker.db')
c = conn.cursor()
def get_seq_by_GeneSymbol(GeneSymbol):
    c.execute("SELECT * FROM link WHERE Gene_symbol=:Gene_symbol",
{'Gene_symbol':GeneSymbol})
    return c.fetchall()
def get_seq_to_predict(GeneSymbol):
    GeneSymbol = GeneSymbol.upper()
    seq_to_predict=[]
    linker = get_seq_by_GeneSymbol(GeneSymbol)
    if linker is None:
         print("no match gene found")
    else:
        print('No. of protein coding transcript found = '+str(len(linker)))
 count = 0
        for line in linker:
            count = count+1
            print('Analysing tanscript ' +str(count)+'......')
            id_e = line[2]
            print('Transcript ID = '+str(id_e))
            server = "https://rest.ensembl.org"
            ext = "/sequence/id/{}?".format(id_e.split('.')[0])
             r = requests.get(server+ext, headers={ "Content-
type" : "text/plain","Connection": "close"})
            if not r.ok:
              r.raise_for_status()
              sys.exit()
            print('Calculating binding sites')
            utr_seq=get_3utr(id_e)
            if utr_seq is not None:
                seq=r.text.replace('T','U')
```

```
if len(seq) > len(utr_seq)*1.5:
                    l=len(seq)-int(len(utr_seq)*1.5)
                else:
                    l= len(seq)-len(utr_seq)
                n= RNA.pfl_fold_up(seq,16,40,80)
                for i in range(l,len(seq)):
                    if n[i][4]>0.2:
                        s = seq[i-23:i]
                        seq_to_predict.append(s[::-1])
            else:
                continue
        return list(set(seq_to_predict))
```

Getutr.py

```
def get_3utr(transcript_id):
    link = ('https://asia.ensembl.org/Homo_sapiens/Export/Output/Transcript?d
b=core;'+  'flank3_display=0;flank5_display=0;output=fasta;strand=feature;'+
 't={}'.format(transcript_id)+';param=utr3;genomic=unmasked;_format=Text')
    utr = requests.get(link)
    utr = utr.text.split('>')[1]
    utr_split = utr.split('\n')
    utr_seq=''
    if 'utr3'in utr_split[0]:
        for fasta in utr_split[1:]:
            utr_seq=utr_seq+fasta.replace('\n','')
         return(utr_seq)
    else:
        return(None)
```

Mirbot_cnn.py

```
def decode_sequence(input_seq):

          states_value = encoder_model.predict(input_seq)
           target_seq = np.zeros((1, 1, num_decoder_tokens))
```

```
     target_seq[0, 0, target_token_index['\t']] = 1.
    stop_condition = False
    decoded_sentence = ''
    while not stop_condition:
       output_tokens, h, c=decoder_model.predict([target_seq] + states_value)
    sampled_token_index = np.argmax(output_tokens[0, -1, :])
        sampled_char = reverse_target_char_index[sampled_token_index]
        decoded_sentence += sampled_char
        if (sampled_char == '\n' or
           len(decoded_sentence) > max_decoder_seq_length):
            stop_condition = True
        target_seq = np.zeros((1, 1, num_decoder_tokens))
        target_seq[0, 0, sampled_token_index] = 1.
        states_value = [h, c]
    return decoded_sentence
def predict_mirna(seq):
    input_seq = seq
    New_encoder_input_data = np.zeros(( 1,max_encoder_seq_length, num_encoder_tokens), dtype='float32')
    for t,char in enumerate(input_seq):
            New_encoder_input_data[0,t, input_token_index[char]] = 1.
 return decode_sequence(New_encoder_input_data)
```

Blast_Predicted_mirna_seq.py

```
def Blast_seq(mirna):

    with open('../data/mirna.fasta','w+') as f:
        f.write('>'+'refseq_1'+'\n'+ str(mirna))
    if os.path.isfile('blast_result.csv'):
        os.remove('blast_result.csv')
    blastx_cline = NcbiblastnCommandline(query='/data/mirna.fasta', db="/data/human_mirna", evalue=0.05,outfmt=10, out="blast_result.csv",word_size= 7,
 gapopen = 50, gapextend = 3, strand= 'both')
    stdout, stderr = blastx_cline()
    list_of_mirna = []
    try:
        with open('blast_result.csv','r+') as f:
            lines = f.read()
            if '\n' in lines:
                lines = lines.split('\n')
            for line in lines:
                if ',' in line:
                    list_of_mirna.append(line.split(',')[1])
        if len(list_of_mirna)>0:
            return list_of_mirna
        else:
            return None
    except:
        return None
```

Has_mir.py

```
def insert_seq(Id,seq):

        c.execute("INSERT INTO hsa_mir_seq VALUES (:Mir_ID , :seq )", {'Mir_ID':Id, 'seq': seq})
def get_seq_by_mir_id(Id):
            c.execute("SELECT * FROM hsa_mir_seq  WHERE Mir_ID=:Mir_ID ",{'Mir_ID':Id})
            return c.fetchall()
def get_all_data():
    c = conn.cursor()
    c.execute("SELECT * FROM hsa_mir_seq")
    return c.fetchall()
```

REFERENCES

1. Ghoshal, A., Zhang, J., Roth, M. A., Xia, K. M., Grama, A. Y., & Chaterji, S. (2018). A Distributed Classifier for MicroRNA Target Prediction with Validation Through TCGA Expression Data. IEEE/ACM transactions on computational biology and bioinformatics, 1051https://doi.org/10.1109/TCBB.2018.2828305
2. J. R. Bertini, V. A. Kasahara and M. C. Nicoletti, "Approaching miRNA Family Classification Through Constructive Neural Networks," 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1-8, doi: 10.1109/IJCNN.2018.8489019.
3. J. Yan, Y. Li and M. Zhu, "miTarDigger: A Fusion Deep-learning Approach for Predicting Human miRNA Targets," 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2020, pp. 2891-2897, doi: 10.1109/BIBM49941.2020.9313504.
4. Karol Szafranski, Molly Megraw, Martin Reckzo and Artemis G.Hatzigeorgiou. (2015).Support Vector Machines for predicting MicroRNA hairpins 33: 831–838.
5. Mohammed Q. Shatnawi & Alhammouri, Mohammad & Mukdadi, Kholoud. (2015). Increasing the Target Prediction Accuracy of MicroRNA Based on Combination of Prediction Algorithms. International Journal of Advanced Computer Science and Applications. 7. 10.14569/IJACSA.2016.070653.
6. N. Sedaghat, M. Fathy, M. H. Modarressi and A. Shojaie, "Combining Supervised and Unsupervised Learning for Improved miRNA Target Prediction," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 15, no. 5, pp. 1594-1604, 1 Sept.-Oct. 2018, doi: 10.1109/TCBB.2017.2727042.
7. S. Cheng, M. Guo, C. Wang, X. Liu, Y. Liu and X. Wu, "MiRTDL: A Deep Learning Approach for miRNA Target Prediction," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 13, no. 6, November 2016.
8. Xueming Zheng, Long Chen, Xiuming Li (2020), “Prediction of miRNA targets from interaction sequences”, https://doi.org/10.1371/journal.pone.0232578
9. Y. Xiao, J. Cai, Y. Yang, H. Zhao and H. Shen, "Prediction of MicroRNA Subcellular Localization by Using a Sequence-to-Sequence Model," 2018 IEEE International Conference on Data Mining (ICDM), 2018, pp. 1332-1337, doi: 10.1109/ICDM.2018.00181.