UNIVERSITÀ DEGLI STUDI DI MILANO
Facoltà di Scienze e Tecnologie
Dipartimento di Informatica

DEEPMIRNA: A NOVEL APPROACH TO MICRORNA TARGET PREDICTION USING DEEP LEARNING

Relatore: Prof. Sebastiano Vigna

Correlatore: Prof. Paolo Boldi

Tesi di:
Simone Sinigaglia
Matricola: 902960

Anno Accademico 2018/2019

to anyone who believed in me . . .

Abstract

This is the abstract

Acknowledgments

This is the ack

Contents

Chapter 1

Introduction

Hola chicos

Chapter 2

MicroRNAs and their importance in living beings

2.1 What are microRNAs?

MicroRNAs (abbreviated miRNAs) are a family of $\approx$ 22-nucleotide small non-coding RNAs that regulates gene expression at the post-transcriptional level [1]. This means that they act by binding to partially complementary sites on target genes, which had been previously transcribed from the DNA of the cell, to induce cleavage or repression of productive translation, preventing this way the target gene to be able to exit the cell and start the translational process that produces peptides and proteins.

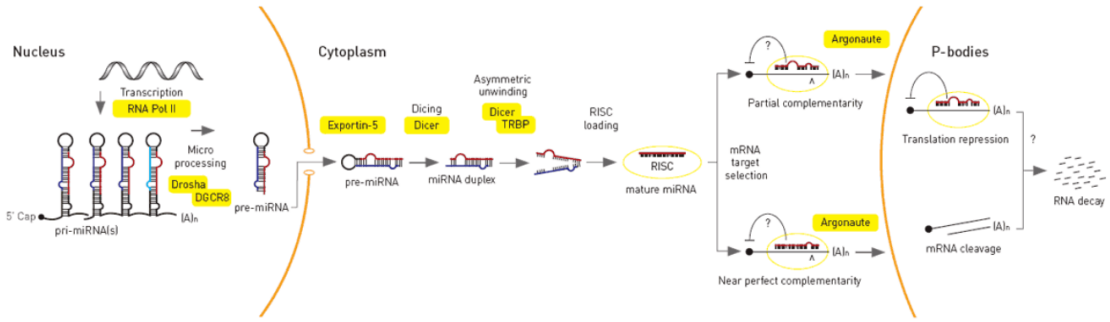

Figure 1: MiRNAs genesis and functionalities

2.1.1 Transcription and processing of miRNAs

As shown in figure 1 miRNAs genes are transcribed by the RNA polymerase II as as large primary transcripts (pri-miRNA) that are processed by a protein complex containing the enzyme Drosha, to form an approximately 70 nucleotide precursor miRNA (pre-miRNA). This precursor is subsequently transported to the cytoplasm where it is processed by a second enzyme, called DICER, to form a mature miRNA of approximately 22 nucleotides. The mature miRNA is then incorporated into a ribonuclear particle to form the RNA-induced silencing complex, RISC, which mediates gene silencing.

It's important to note that, generally, only one of the two strands of the stem loop is incorporated into the silencing process, and it's selected on the basis of its thermodynamic instability and weaker base-pairing on the 5′ end relative to the other strand. The latter, called the passenger strand due to its lower levels in the steady state, is usually denoted with an asterisk (*) and is normally degraded. However, in some cases, both strands of the duplex are viable and become functional miRNAs that target different mRNA populations. (see figure 2)

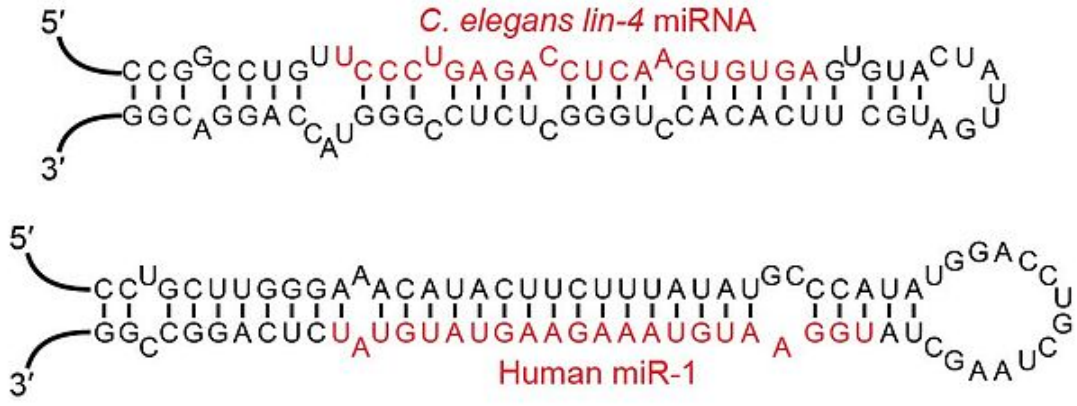

Figure 2: Examples of miRNA stem-loops. In red is shown the mature miRNA

2.1.2 RNA-induced silencing complex (RISC)

As mentioned before the mature miRNA is part of an active RNA-induced silencing complex (RISC). This process represents their main functionality in both animals and plants. The RISC it is a key process in gene silencing and can act in two

different ways as depicted in the right-hand side of picture 1: via mRNA degradation or by preventing mRNA translation. It has been demonstrated that given complete complementarity between the miRNA and target mRNA sequence, Ago2 can cleave the mRNA and lead to direct mRNA degradation. In the presence of only partial complementarity instead, silencing is achieved by preventing translation [2].

2.2 Why are they important?

MiRNAs are particularly abundant in many mammalian cell types and appear to target about 60% of the genes of humans and other mammals [3].

Many miRNAs are evolutionarily conserved, which implies that they have important biological functions [3]. For example, 90 families of miRNAs have been conserved since at least the common ancestor of mammals and fish, and most of these conserved miRNAs have important functions.

The discovery of the first miRNA over 20 years ago has ushered in a new era in molecular biology. There are now over 2000 miRNAs that have been discovered in humans and it is believed that they collectively regulate two third of the genes in the genome.

The repressive action of miRNAs has a huge impact on many biological processes such as cell cycle control and several developmental and physiological processes including stem cell differentiation, cardiac and skeletal muscle development, neurogenesis, insulin secretion, cholesterol metabolism, aging, immune responses and viral replication. [4]

In addition to their important roles in healthy individuals, microRNAs have also been implicated in a number of diseases including a broad range of cancers, heart and neurological diseases. In fact it has been discovered that their expression patterns are highly specific in respect to external stimuli, developmental stage or tissue and this can be used to diagnose diseases in which the expression levels of miRNAs are known to change considerably [5]. Consequently, miRNAs are intensely studied as candidates for clinical diagnosis and predictors of drug response [6].

2.3 Why miRNAs target prediction is a difficult task?

MiRNA's targeting is a complex mechanism, yet not fully understood. This process involves the generation of complex regulatory networks and understanding mechanisms and functions of these networks requires systematic experimental investigation. In the ideal world it would be possible to experimentally validate

every single of all miRNAs, but the cost and time needed for the verification make miRNA studies still depending on computational predictions to complement experimental data.

The apparent complementarity between miRNA and its target could be seen as an advantage for computational analysis; however, other features of miRNA-UTRs association make matters more complicated. The usual sequence alignment algorithms assume longer sequences than the 20-23nt of miRNAs. This short length makes ranking and scoring of targets very difficult as statistical techniques for sequence matching require longer sequences to be significant. Besides, binding sites actually consist of regions of complementarity, bulges and mismatches. Hence, complementarity is not sufficient to identify functional targets.

Recent studies [7] [5] reveal that a single miRNA can bind to many different genes. On average, each miRNA have interactions with over 200 mRNAs and these genes may offer tens of different site locations along their sequence, however, not all these interactions give origin to a functional binding site: the same miRNA:mRNA pair may create a bond which is functional in a certain tissue but not inside another cell type.

Besides, environmental conditions and other external factors may also contribute to binding sites functionality as described in [8].

All these circumstances make identification of miRNA targets a very challenging task that must be carefully tackled in order to have a better understanding of their role in the gene silencing complex.

Chapter 3

MicroRNAs target prediction computational methods

3.1 Introduction

Earlier in Chapter 2 we described how miRNAs play a fundamental role in gene regulation. It is common belief that the final and probably most relevant step in their regulatory pathway is targeting [5]. Targeting is intended as the binding of the mature miRNA to the messenger RNA via the RNA Induced Silencing Complex (see figure 3). Hence, valid targets need to be identified for miRNAs in order to properly understand their role in cellular pathways.

However, many of the discovered miRNAs do not yet have identified targets. This is especially the case in animals where the miRNA does not bind to its target with a nearly perfect matching as it does in plants [9]. Experiments have proved that a single miRNA has the potential to regulate hundreds of target mRNAs and multiple miRNAs may compete for the regulation of the same mRNA [10], however target validation is difficult, expensive, and time consuming. Thus, having considered all these facts, it is of crucial importance to have accurate computational miRNA target predictions.

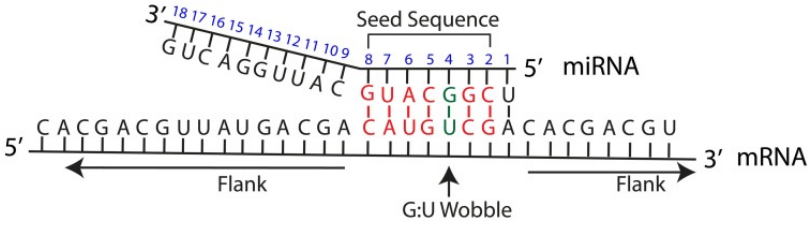

Figure 3: Example of miRNA targeting.

3.2 MiRNA target prediction

Before miRNA target prediction tools were available, possible miRNA binding sites were determined manually. These target sites were later confirmed by laborious and inefficient techniques such as site-directed mutagenesis and other experimental methods. The identification of the first targets for the let-7 and lin-4 miRNAs led to the idea that miRNAs have a pattern in targeting genes which could be used to develop target prediction algorithms [11].

Originally gene targeting by miRNAs was believed to be the result of their binding to the 3'UTR of the target mRNA [10], however recent studies [12] have confirmed gene regulation as a result of the binding of the miRNA to the coding region as well as to the 5'UTR. Furthermore, computational evidence suggests that regulation via the binding of the miRNA to the coding region differs in comparison to the binding pattern seen at the 3'UTR. In particular, it's suggested that miRNAs target the coding regions of mRNAs with short 3'UTRs [13].

Another key factor in target prediction is that 3'UTRs are prone to change under different conditions which might result in the elimination of the target site. Binding in the coding region on the other hand may instead present an evolutionary advantage for the cell as it could help in the preservation of the miRNA binding site [14].

3.2.1 Features and methodologies

While many miRNA targets have been computationally predicted only a limited number have been experimentally validated. Moreover, although a variety of miRNA target prediction algorithms are implemented, results amongst them are generally inconsistent and correctly identifying functional miRNA targets remains a challenging task.

The average performance of target prediction tools, which typically identify approximately 80% of known miRNA targets, indicates that the mechanisms associated with miRNA-regulated processes remain poorly understood. Thus, there is a room for novel approaches to improve the knowledge of the rules that govern their targeting process [15]

The various methodologies implemented use several different approaches and analyze a wide range of features for this task. Almost all target prediction methods are rule-based or adopt machine learning methodology with varying success. Rule-based systems incorporate various human-crafted descriptors to represent miRNA:gene target binding (e.g. type of pairs in the site, binding stability, or conservation of the target site among species). Machine learning techniques also use those descriptors, but as input features to machine learning models. The limitation

of both these approaches is indeed the process of feature selection and representation, which is constrained by the use of human selected descriptors to model a process that is not fully understood.

The most common characteristics used in miRNA targets identification are [7]:

- seed region complementarity
- free energy
- site accessibility
- conservation

The seed region

Targeting patterns are very different between plants and animals. Plants, in fact, show a near perfect complement between their miRNA and the respective target mRNA. On the other hand animal miRNAs bind their targets with only partial complementarity. In particular, a region of about 6 to 8 nucleotides in length at the beginning of the miRNA is of crucial importance in the targeting. This short subsequence is called *seed region* and it comprises the nucleotides between the second at the eighth (the seed sequence in Figure 3) starting from the 5′ end. The seed region is very important because it binds to the target mRNA leading to the regulation of the gene in question [16].

Undoubtedly the seed region is one of the most commonly used miRNA traits for target prediction. This seed-centric view, in fact, has been supported by structural studies [17] and a widely cited report [18] that investigated the importance of other (non-canonical) regions within a miRNA concluding that their contributions had relatively low relevance compared to the (canonical) seed region. More recent experiments, however, have highlighted a role for the entire miRNA, suggesting that a more flexible methodology is needed [19].

Free energy

The free energy, also called hybridization energy, is defined as the energy released by the pairing between the miRNA and mRNA and it can be used as the measure of the stability of the bond. In fact a stable bond is considered more likely to be a functional target of the miRNA. However, since measuring this quantity directly is difficult, usually the change of free energy during a reaction is considered (ΔG). Reactions with a negative ΔG have less energy available to react in the future, hence they result in systems with an increased stability. By predicting how the miRNA and its candidate target hybridize, regions of high and low free energy

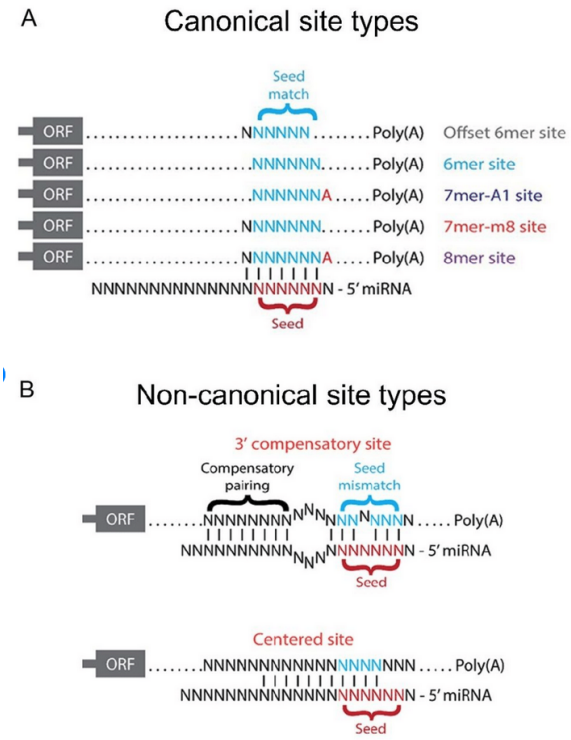

Figure 4: Example of canonical and non-canonical binding sites.

can be inferred (Figure 5) and the overall ΔG can be used as an indicator of how strongly bound they are [20]. The lower this value the more stable the binding.

Figure 5: A hairpin loop is shown with the loop corresponding to a region of high free energy (a positive ΔG) and the stem corresponding to a region of low free energy (a negative ΔG)

Site accessibility

Site accessibility is the measure of the ease with which a miRNA can locate and hybridize with its target. After transcription, in fact, a mRNA assumes a certain secondary structure which can interfere with the miRNA ability to bind to its target site. To understand why this is important, we need to consider that the miRNA:mRNA hybridization involves a two-step process in which a miRNA firstly binds to a short accessible region of the mRNA and only after, while the secondary structure of the mRNA unfolds, completes the binding. It is likely that secondary structures contribute to target recognition, because there is an energetic cost to freeing base-pairing interactions within mRNA in order to make the target accessible for miRNA binding. Hence, to assess the likelihood that a mRNA is a target of a given miRNA, the predicted amount of energy required to make the site accessible (the so called site accessibility energy SAE) should be taken into consideration [21].

The SAE can be computed as the difference between the free energy cost of opening the mRNA and free energy gained from the intermolecular interaction (Figure 6).

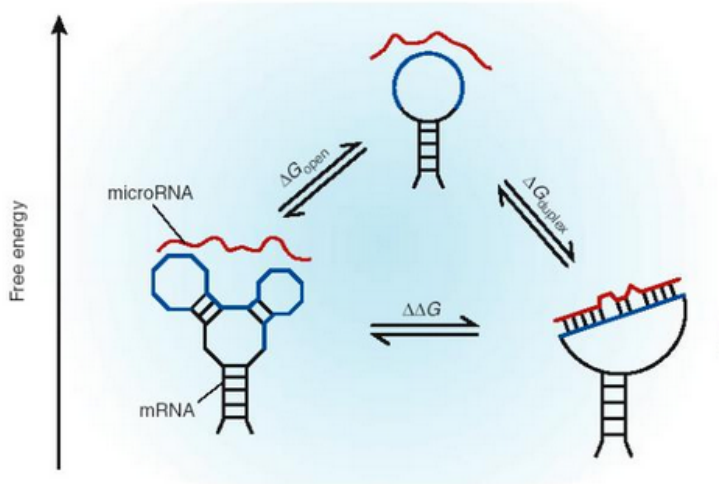

Figure 6: Binding of a miRNA to its target mRNA is depicted as a 2-step process. Portion of the mRNA structure must be open before miRNA:mRNA base pairing can be established.

Conservation

Conservation refers to the maintenance of a sequence across species. According to many reports [5] looking at conserved targets between different species helps reducing the number of false positive results. However, other more recent studies highlighted the fact that this may also increase the number of false negatively identified targets [3].

Chapter 4

Experimental setup

4.1 Introduction

The majority of prediction tools are based on the assumption that it is the miRNA seed region that contains almost all the important interactions between a miRNA and its target and their focus are on these canonical sites.

In this thesis, we aim at using deep learning techniques to investigate the role of those non-canonical sites and pairing beyond the canonical seed region in miRNA targets. Recent increases in computational power have permitted the rise of methods that can dispense with human-crafted features, making it possible to deal directly with raw data and autonomously learn and identify patterns to appropriately represent it. In particular, deep learning has been shown to be an effective method for classification tasks in domains with complex feature representation [22].

Deep Learning (DL) has already been applied to the miRNA target prediction problem. Cheng et al. [23] used convolutional neural networks to analyze matrices of miRNA:site features, but the selected features were still human-crafted descriptors and thus the method faces similar problems as rule-based and ML approaches. A more recent work, DeepTarget [24], relied on recurrent neural networks to identify potential binding sites and assess their functionality. However this work is still oriented to the identification of canonical sites and relies on a limited small data set for the training phase. Another approach using DL is DeepMirTarSdA [25], that explore the use of stacked de-noising auto-encoders (SdA) to predict human miRNA-targets at the site level, but the network obtained is huge and the small availability of input data (about 8000 samples between positives and negatives) results in a model that performs well on the data being used but generalizes quite poorly.

In this thesis we present DeepMiRNA, a miRNA target prediction tool that attempts to exploit the learning capacity of a neural network to extract abstract patterns from raw input data. Unfortunately, to the best of our knowledge, there is no suitable raw-data representational method for miRNA-target prediction. This is very likely the consequence of the very small quantity of validated data available for this task. For this reason, rather than making assumption about suitable descriptor, we created a set of rules to help finding the best candidate binding sites leaving the classification decision to the neural network (see CSSM in the next section).

More precisely, DeepMiRNA scans the 3'UTR of the gene identifying potential target sites according to the chosen rules. It then uses the previously trained network to identify the relevant patterns by directly examining the whole mature miRNA transcript, rather than focusing on the seed region and analyzing precomputed descriptors. In the following text we denote a miRNA binding site with MBS.

4.2 Preparing for training

Two of the fundamental properties in deep neural network theory states that:

1. with sufficient data samples and a correct network design a NN can approximate any mathematical function

2. a NN has the capacity to automatically learn the relevant features of complex data structures by means of its hidden layers [22]

For these reasons, in our approach we sought to minimize potential biases introduced by handcrafted features by working with the miRNA and the mRNA transcripts, feeding them directly to the neural network.

The DeepMiRNA working pipeline for the identification of functional targets for miRNAs can be summarized as follows (Figure 7: a 30-nucleotide sliding window with stride of 5 nucleotides is used to scan the 3'UTR of a given gene. Both values (size and stride) has been empirically computed during the training stage (see the results section for more informations). For each mRNA 30-nucleotide long subsequence, the VIENNA RNACofold package [26] is used to compute the stability of the binding between the miRNA transcript and the fragment. If the computed value is below a predetermined threshold, the primary structure of the mRNA and the miRNA are examined to see if the criteria defined by the candidate site selection rules (CSSR) are met. If so, the duplex is vectorized and fed into the

network for classification. The prediction is then further refined using an a posteriori filter that computes the site accessibility of the mRNA region surrounding the predicted binding site.

In the next two sections we will describe how DeepMiRNA's neural network actually comes in two different flavors according to the encoding algorithm used: either one feed-forward or one convolutional network.

We pointed this out because, despite the important role of site accessibility in miRNA target identification described in recent articles [21] [7], the filtering stage has only been used with the convolutional neural network. This decision has been made according to the experimental results obtained: the feed-forward model in fact, exhibited a great ability in identifying negative binding sites correctly. The filtering step does not significantly increase the model's performance while it slows down the testing process, hence we opted for its exclusion.

The situation changes when the convolutional network is employed for the task: for this model site accessibility has revealed an important role in false positive reduction, justifying the decision to keep the filtering stage with this configuration. More details about this feature can be found in section 4.3.3.

4.2.1 Data preprocessing

A key factor for successful application of any Machine Learning technique and one of the most important aspect to consider before feeding the network with data is how we prepare the input data and the targets.

The main aspect to consider during this process is the use of suitable techniques to make the data more amenable for the NN: this includes vectorization, normalization and handling of missing values.

vectorization

In order to be accepted by the neural network all inputs must be tensors, that is they must be expressed by numerical arrays with suitable shape and dimension. The process to encode categorical data, in our case the transcript sequences, into tensor is called *vectorization*. In this thesis we actually employed two different techniques for this operation:

- one hot encoding of the sequences [27].
- sequence embedding using a Word2Vec approach [28].

One hot encoding is a process by which categorical data such as strings are converted into binary vectors. In our case, each nucleotide is translated to a binary

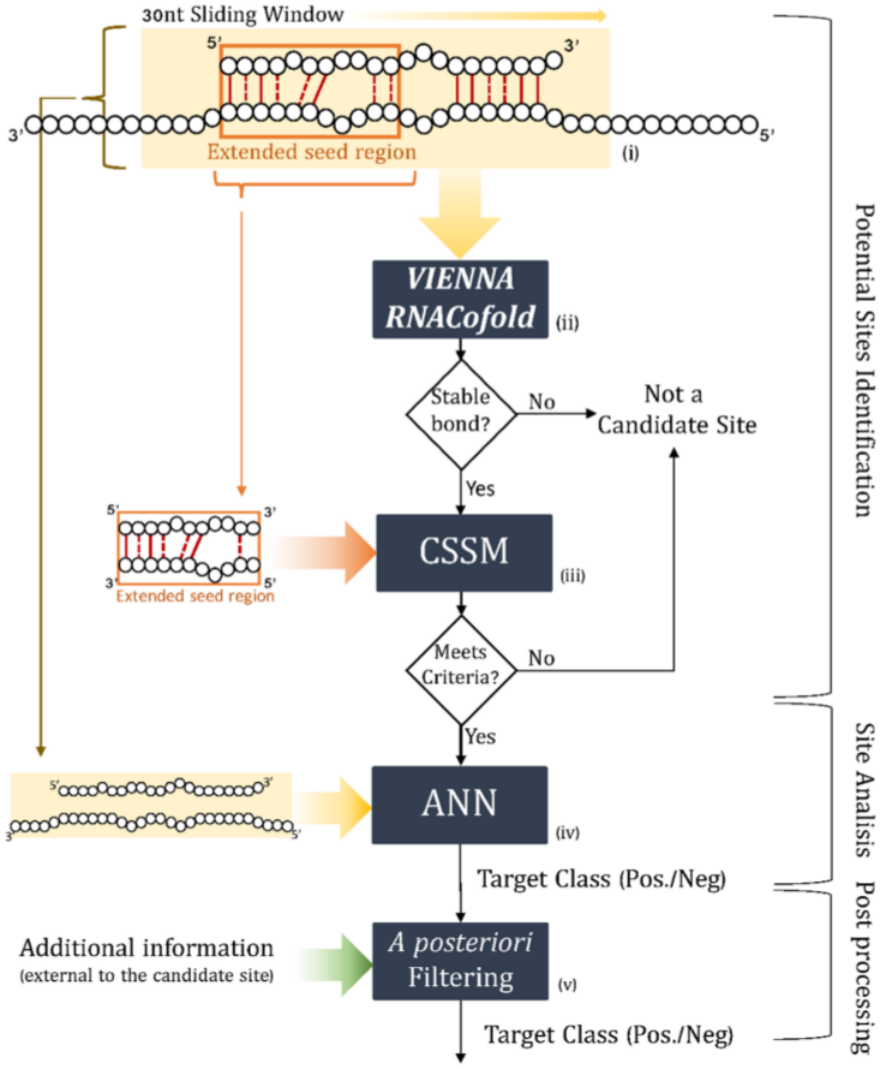

Figure 7: **DeepMiRNA pipeline**. (i) a 30nt sliding window with a 5nt step is used to scan the gene transcript; (ii) The Vienna Cofold software is used to compute the binding stability; (iii) if the bond is predicted to be stable a partial complementarity according to the rules defined in the CSSR is verified; (iv) if all previous checks passed the duplex is fed to the NN for prediction; (v) for positive predictions a filter is used to compute the site accessibility of the miRNA: if the energy needed to access the site is above a certain threshold the prediction is changed to negative.

vector of size 4, corresponding to the four possible nucleotide values as described in Table 1

The main problem using this method is that not all duplexes have the same size.

This is in particular due to the different miRNA's transcript length (ranges from 18 to 30), The network, instead, requires that all inputs have the same shape. Hence, in order to meet this requirement, every miRNA sequence, when needed, has been padded with 'empty' letters to reach the maximum size length (in this case 30). Regarding the site transcript, each fragment has size 40: 30 corresponding to the window size plus 5 additional nucleotides upstream and downstream. These additional nucleotides seek to capture any influence that the flanking sequence may exert on the target [3]. With these adjustments each duplex is represented by a binary vector of (fixed) size 280.

The second vectorization method uses a Word2Vec approach. Whereas the vectors obtained through one-hot encoding are binary and sparse (mostly made of zeros), sequence embeddings are usually dense, very low-dimensional floating-point vectors.

Word2Vec [28] is a Natural Language Processing (NLP) methodology to map words into numeric vectors based on their context. Being the context defined as the words surrounding the word to encode. For DNA sequences, however, there is no clear definition for words, so usually a k-mer (that is a set of k continuous nucleotides)is used to define a word (see Figure 8). Therefore, in case of biological sequences the context is defined as the set of n adjacent k-mers (being n a parameter to validate). For this thesis use the software available at `https://github.com/pnpnpn/dna2vec` to train the model used to encode the k-mers. This encoding has two important advantages compared to one hot encoding:

1. each k-mer of length comprised between 3 and 8 is mapped to an equal size vector of size 100.

2. similar k-mers are mapped to close points in the features space according to a specific distance metric (usually Euclidean distance).

In our case each variable lenght miRNA sequence has been split into 4 different size k-mers each mapped into a 100-dimension vector, while each fixed size site

Table 1: One Hot encoding of a nucleotide.

Nucleotide	**Encoding**
A	[1, 0, 0, 0]
C	[0, 1, 0, 0]
G	[0, 0, 1, 0]
U or T	[0, 0, 0, 1]
Empty	[0, 0, 0, 0]

transcript (plus the flanking nucleotides) has been split into 5 8-mers. This way each duplex is mapped into a $9x100$ matrix obtained concatenating the resulting 9 vectors. It's important to note that this vectorization requires a different design and implementation of the neural network to use as we will describe in the next section.

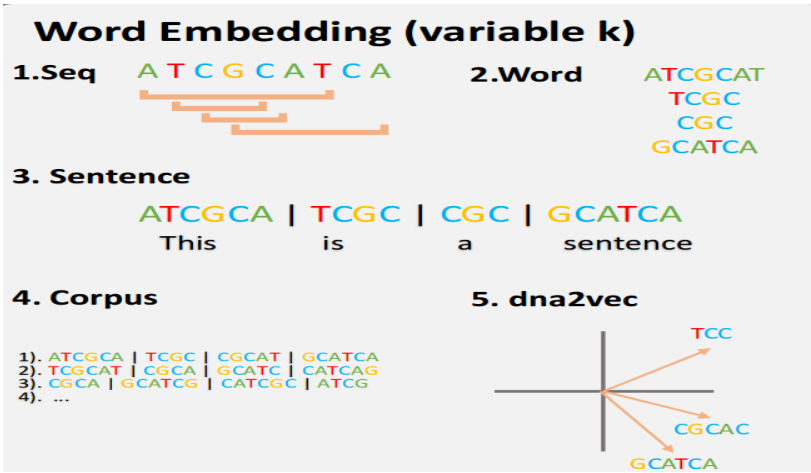

Figure 8: Dna2Vec mapping example using different length k-mers

normalization and missing values

Another crucial part of data preprocessing concerns their normalization. In general, it's not safe to feed the network with data that takes relatively large values or that are heterogeneous (i.e have very different values ranges). In our case, however, the vectorization process guarantees that the numerical values resulting from the encoding are both small and homogeneous. Regarding missing or incomplete data, we found a very small quantity of them in the datasets retrieved, hence we simply decided to discard them.

4.2.2 Dataset preparation

One of the biggest challenge to face in any Machine Learning application is to get a good set of significant data. The main purpose is to have a sufficiently variable and representative dataset that allows the model to generalize well on new and unseen examples. This phase is probably the most important together with the network design as it's crucial for achieving good performances. It is of fundamental importance that the data is representative for the problem we want to solve. It does not matter how big is the quantity of data we obtain if that is not aligned with the goal

Table 2: Example of Diana TarBase entries.

gene name	miRNA name	functionality
AB002442	hsa-miR-192-5p	POSITIVE
DAD3R	hsa-miR-34a-5p	NEGATIVE
E2IG4	hsa-miR-200c-3p	POSITIVE

we want to pursue. One should focus on finding data with features that matter to what we're trying classify or predict and discard unrelated features. Hence, the first step of a classification process should be proper data collection, and until we achieve this, we will find ourselves constantly coming back to this step [29].

For the miRNA target prediction task, this requires a comprehensive dataset of verified positive and negative targets that encompass both canonical and non-canonical examples. While there are multiple data repositories providing information regarding experimentally validated positive miRNA targets[30] [31] there are significantly fewer experimentally verified negative targets. This represents a major concern for ML-based approaches that require similar numbers of labeled examples for both classes.

In this thesis we only considered human data and we used the version 8 of the Diana TarBase[1] and the release 7.0 of miR TarBase[2] datasets. Moreover, in order to annotate the datasets with lacking informations such as molecules transcripts, we downloaded gene sequences from Ensembl[3] and miRNA transcripts from miRBase[4]

Diana TarBase

The Diana TarBase is the most widely used database for validated miRNA interactions. The latest version at time of writing is $v8$ and asking for a bulk download from their website is possible to obtain a dataset of 526658 positive and 63559 negative validated targets. This dataset provides informations about miRNA:mRNA interactions in the form of miRNA name, gene name and functionality as it's illustrated in Table 2. Unfortunately there is no information about transcript sequences of the molecules or binding sites location.

[1] `http://diana.imis.athena-innovation.gr/DianaTools/index.php`
[2] `http://mirtarbase.mbc.nctu.edu.tw/php/index.php`
[3] `https://ensembl.org/index.html`
[4] `http://mirbase.org`

Mir TarBase

This database contains interactions of about 2000 miRNA and around 20000 genes. In total the downloaded dataset comprises 380091 positive and 382 negative duplexes available from the Download section of the website. It's possible to find it under the name *hsa_MIT.xlsx*. It's important to note that every single miRNA can interact with hundreds of different genes. Again, exactly as with the Diana TarBase dataset, no sequence or binding sites transcripts are provided.

Ensembl

From Ensembl[32] is possible to obtain the transcripts of all human genes. In this thesis we focused on the 3'UTR and for the download we used the BioMart tool available from the main menu. More specifically, from the BioMart webpage we followed these steps:

- from the drop down menu choose database: *Ensembl Genes 96*;
- then choose dataset: *human genes GRCh38.p12*;
- from the left-hand side of the page click attributes and select *sequences*;
- from this submenu select *3'UTR* and then click on *Header Information*;
- now from the header submenu select the following attributes: *Gene stable ID*, *Gene name*, *Transcript stable ID*, *3'UTR start* and *3'UTR end*;
- lastly click on *Results* from the top left and export the generated fasta file selecting the box *Unique results only* and pressing *Go*.

Note that a single gene identified with a unique ID may produce multiple RNA's transcripts each denoted with its own transcript ID. The actual transcript observed will depend on on the tissue, developmental time point, and environmental or hormonal factor. Typically there's a single major transcript expressed in a given cell at a given time, but not always. Unfortunately in both the Diana TarBase and miR TarBase the transcript ID of the miRNA:mRNA pair is not present, hence, according to what is used in [25], we kept the transcript with the longest sequence.

mirBase

The mirBase database allowed us to retrieve all known miRNAs transcripts. For the homo sapiens species there currently around 2000 miRNAs sequences and in order to collect them we followed these steps:

- from the homepage click *Browse* from the top menu;
- click on *human* and a preview of the data will appear on screen as depicted in figure 9;
- on the bottom of the page select sequence type *Mature sequence* and output format *Unaligned fasta format*;
- press *Select all* and then *Fetch sequences*;
- the query result will be printed on the next page where it can be copied and pasted to a regular text file and saved as a fasta file.

Homo sapiens miRNAs

ID	Accession	RPM	Chromosome	Start	End	Strand	Confidence
hsa-let-7a-1	MI0000060	145261	chr9	94175957	94176036	+	✓
hsa-let-7a-2	MI0000061	142652	chr11	122146522	122146593	-	✓
hsa-let-7a-3	MI0000062	142757	chr22	46112749	46112822	+	✓
hsa-let-7b	MI0000063	83403	chr22	46113686	46113768	+	✓
hsa-let-7c	MI0000064	135622	chr21	16539828	16539911	+	✓
hsa-let-7d	MI0000065	4649	chr9	94178834	94178920	+	✓

Figure 9: A sample of the mirBase database.

As a preliminary step, the Diana and Mir TarBase data were parsed to remove inconsistent entries, that were marked both as positive and negative targets due to contradictory results in different experimental validations, and combine entries that were validated more than once by different verification methods. We also filtered pairs were the annotated 3'UTR sequence length was shorter than 200 nucleotides. This produced a final dataset of 593407 positive (+) and 33604 negative (-) miRNA:mRNA interactions containing bindings for about 16000 different genes and around 2000 miRNAs. This data was then split into two parts for the training and testing phases using a ratio of 67:33, thus obtaining a training functional set of 397582 positives and 22504 negatives and a test set of 195825 positives and 11100 negatives.

4.2.3 Training dataset

Structuring a proper training dataset is an essential aspect of effective deep learning models but one that is particularly hard to solve. Part of the challenge comes from the intrinsic relationship between a model and the corresponding training dataset. If the performance of a model is below expectations, it is often hard to determine whether the causes are related to the model itself or to the composition of the training dataset.

The purpose of this stage is to train and validate the neural network that has the responsibility for distinguishing between functional (positive) and non-functional (negative) target sites. A positive experimentally validated gene can exhibit tens of potential positive binding sites but not necessarily all them are actually functional. Hence, the training set must be composed of miRNA:MBS pairs rather than miRNA:mRNA duplexes. However, the retrieved data, while consistent and sufficiently various, do not contain such informations. Searching the Internet we were able to find only 2 publicly available validated datasets providing information regarding experimentally identified miRNA binding site locations: the Helwak [19] and the Grosswendt [12] datasets.

Positive binding sites

The two above datasets contain miRNA:MBS locations obtained through PAR-Clip [33] and CLASH [34] experiments, however the binding site identified might not be functional. Hence, in order to consider a site as positive we cross-reference them with the Diana TarBase and miR TarBase. In particular we considered a given duplex miRNA:MBS as functional if:

- form a stable bond, that is it has a free energy below a predetermined threshold accoriding to Vienna Cofold. The threshold used was $-10\ kcal/mol$.+
- correspond to a miRNA:gene pair marked as functional in either the Diana or the miR dataset.

Additionally, we complemented our positive training dataset by including the most probable broadly conserved sites obtained from TargetScanHuman 7.1 [35] that matched experimentally validated functional data from Diana Tarbase or mirTarBase. This process produced a total of 32774 positive miRNA:MBS duplexes for the training stage.

Negative binding sites

The small number of non-functional experimentally validated binding sites makes the construction of a representative negative dataset a very difficult task. Some

of the examined existing tools overcome this issue by creating 'mock' targets [36] continuously shuffling the sequence of a real functional binding site until the resulting transcript does not appear in any positive miRNA target repository. This method, however, may lead the neural network to learn the function used to create the negative examples resulting in a model trained to discriminate between artificial and real data rather than discerning functional targets from non-functional. Besides, there is no guarantee that the generated sequence is indeed a true negative because it has not been experimentally validated.

The solution we implemented opted for using the validated miRNA:mRNA pairs to extract the negative binding sites and add them the ones provided by the CLASH and CLIP experiments. The idea is that any sequence of approximately 30 nucleotides in the mRNA of a negatively validated miRNA:mRNA pair may represent a potential negative target site. However, in order to avoid the introduction of noise including such sequences, we decided to keep only subsequences of mRNA where the associated miRNA has the potential to form a stable bond. In order to check this requirement we used a sliding window of 30 nt along the entire 3'UTR region. For each negative miRNA:mRNA pair we kept the MBS with the lowest binding energy. For some pairs there might be no candidate satisfying the CSSM reequirements, in this case we just discard the entry. From the 22504 negative pairs of the original training set we were able to create 22205 negative MBSs (see Figure 10). The binding energy (or free energy) was computed according to the RNACoFold function from the RNA library [26].

Training stage

Once the positive and the negative datasets have been created, we shuffled and merged them to create a unique dataset. The whole process of creation of the training set is illustrated in figure 10. In order to train and validate the network we kept 80% of the data for training and the rest for validation and test set.

4.3 The testing stage

There is a big difference regarding the procedure adopted for the testing stage compared to the training phase. Here the purpose is to predict if a given miRNA targets a certain gene, hence the testing data consists of pairs containing the miRNA and the whole 3'UTR transcript, rather than a specific MBS. This step is essential to be able to evaluate the whole pipeline process of DeepMiRNA.

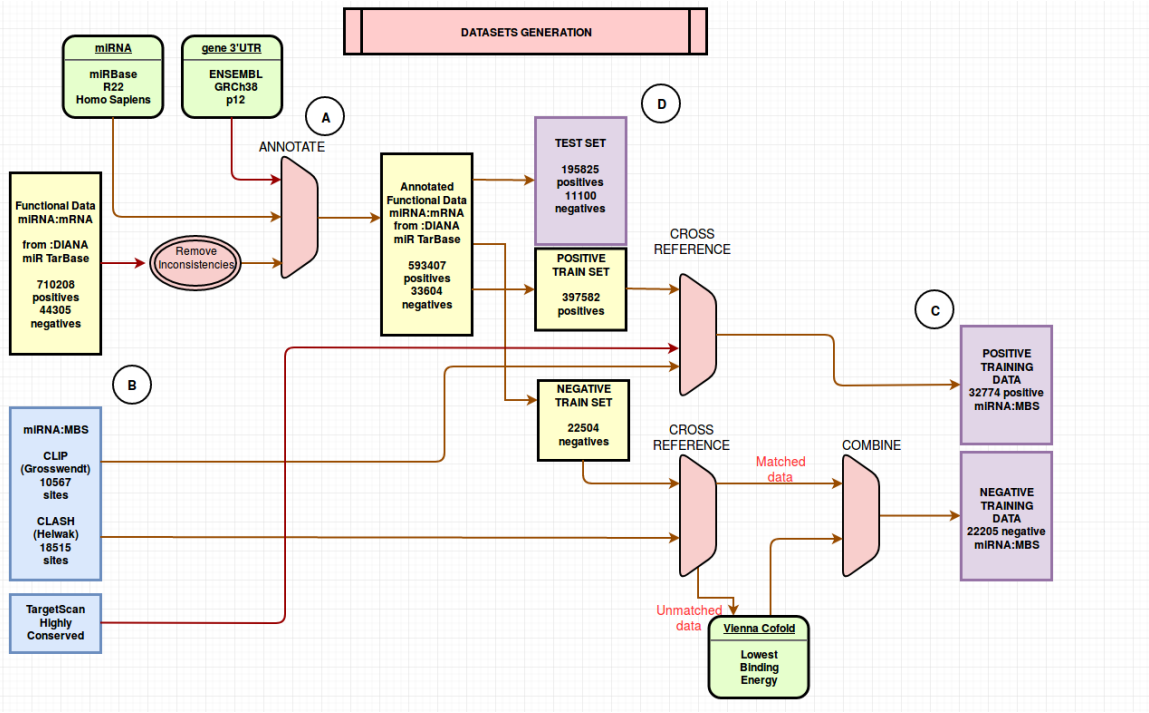

Figure 10: **The datasets generation process.** Two types of data were used: (i) experimentally verified functional datasets that define if a miRNA targets a gene - in yellow - and (ii) CLIP datasets that define miRNA binding site locations in the 3'UTR of a gene - in light blue -. In purple the final test and training sets. (A) Diana e miR TarBase were used as the main source of data: after removing inconsistent and duplicated data we annotated it using the Ensembl and the miRBase databases and subsequently we split it into test and training set. (B) Training data were cross referenced with CLIP and CLASH validated data providing information about the actual binding site of functional targets. The positive set was then complemented using highly conserved MBS locations provided by TargetScan (C) In order to obtain a consistent set of negative data for the training phase, we selected the miRNA:MBS with the lowest binding energy from the the negative train set. (D) The final test set was extracted keeping the 33% of all available data.

To correctly evaluate the network performance we used the experimentally verified miRNA:mRNA pairs excluded from the training set. Those data points, however, were highly biased towards positive entries in a ratio of 95:5 and this imbalance could impede a true evaluation of the trained model. In fact, a tool that exclusively predicts positive targets against the full test data would achieve an accuracy of 95%.

Nevertheless, using suitable metrics, such as f1-score, precision and recall, as

described in [37], is still possible to effectively overcome this issue.

4.3.1 Candidate site selection methods: CSSM

Often, in the academic world, people tend to affirm that, in order to properly train a neural network, one needs to have a huge amount of data available. While we partially disagree with this common belief, convinced that quality and variety prevail over quantity, we still agree that in the case of miRNA targets prediction the available experimentally validated data is still not sufficiently representative of the task and the candidate site selection step effectively narrows the search space to simplify the NN classification task.

For this reason, the selection of candidate sites in a mRNA becomes a key step for a miRNA target sites predictor because it helps identifying which regions within the mRNA have the potential to accommodate a binding site. We found out that most of the publicly available algorithms follow a similar approach: they scan the gene's 3'UTR looking for sites that are partially complementary to the miRNA transcript; if a site meets certain criteria, it is considered to be a candidate site and is subjected to further analysis.

In the light of new recent results, stating that functional miRNA targets can arise either by a single strong binding site (i.e. 6 consecutive complementary nucleotides) or by multiple weak binding sites [19], we believe it's essential to consider the whole 3'UTR sequence using CSSM willing to accept both canonical and non-canonical sites. These methods are less conservative and allow accepting bulges, mismatches or wobble pairs in the seed region (see figure 11).

The CSSM adopted in DeepMiRNA use a very similar approach to miRAW [38]: we consider a 30-nucleotide window to scan the 3'UTR and we extend the typical 7mer seed region (figure 11a) considering the first 10 nucleotides of the miRNA transcript (figure 11d) to look for partial complementarity.

In particular, we consider a site to be a potential candidate site if there is a minimum number of base pairs, both Watson-Crick and Wobble, within the extended seed region. In this thesis we investigated three different configurations:

1. CSS-6.0:10: a candidate site must contain at least 6 base pairs between the extended seed region composed of the first 10 nucleotides;

2. CSS-7.0:10: a candidate site must contain at least 7 base pairs between the extended seed region composed of the first 10 nucleotides;

3. CSS-7.1:10: a candidate site must contain at least 7 base pairs between the extended seed region composed of the 9 nucleotides comprised from the second and the tenth.

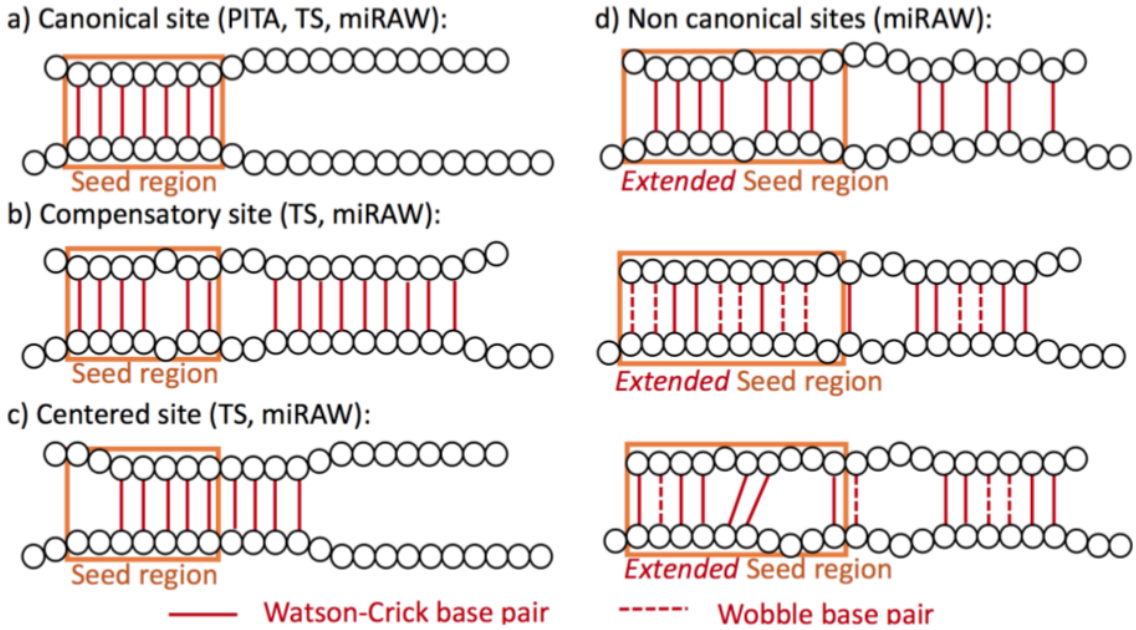

Figure 11: Example of canonical and non-canonical sites used in three available tools: Pita, TargetScan and miRAW.

In each case, base pairs do not need to be consecutive in order to accommodate the presence of gaps and bulges. The task of finding non-consecutive complementary nucleotides can be mapped to the problem of finding the longest common subsequence (LCS) between two strings as illustrated in figure 12.

Those configurations accept both standard canonical MBSs as well as a broader range of non-canonical target site structures including the vast majority of experimentally validated sites from Diana TarBase and CLIP/CLASH binding site datasets. Moreover, while these relaxed conditions for the seed region generate a much larger number of candidate sites and potentially an increased quantity of false positives, the decision of whether a site represents a functional target is delegated to the neural network. This way, we ensure that minimal assumptions, and hence bias, are incorporated into the analysis.

4.3.2 Longest Common Subsequence implementation

The computational cost of the testing stage is considerably higher than that of the training phase. The reasons lie mainly in the computation of the potential candidate sites, involving check of complementarity and calculation of the minimum free energy, and the cost of the filtering stage (see next section). While there is very little to do with the calculation of the interaction energy of the duplexes, which

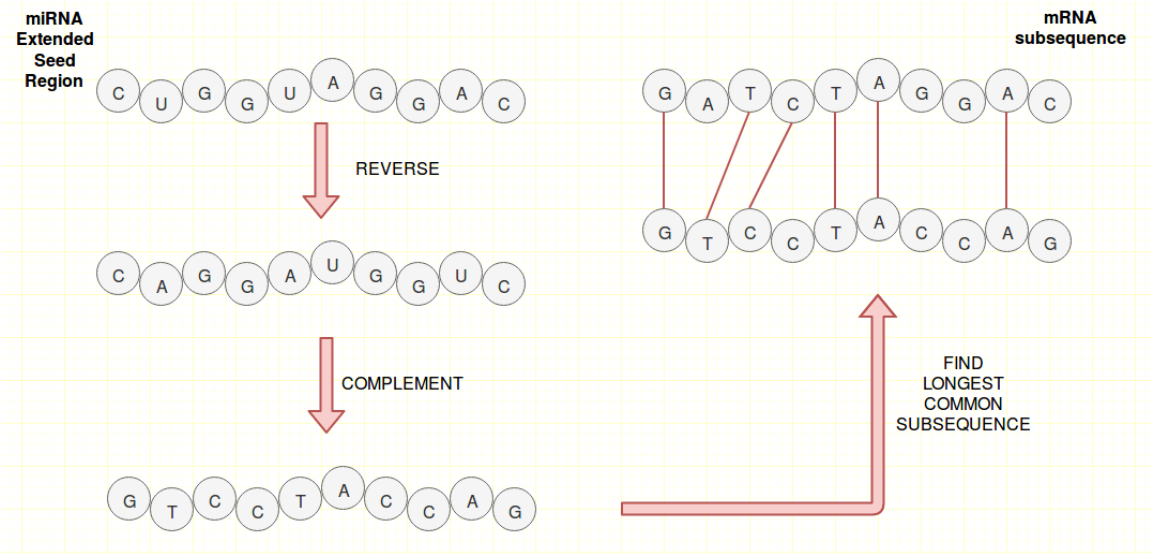

Figure 12: Mapping partial non-consecutive complementarity to longest common subsequence search.

completely relies on the Vienna Cofold library, for the complementarity check we tried different approaches to reduce the computational cost.

As mentioned above the seek of non-consecutive complementary nucleotides between two sequences can be seen as a special case of the problem of finding the longest common subsequence (LCS) between two strings.

The LCS version we implemented in this thesis is done using *Cython*[39]. Cython is a programming language that can be seen as a superset of Python that aims to give C-like performances with code that is mostly written using native Python.

In fact, Cython actually works by producing Python modules where the code, however, is first translated and compiled into C.

For this reason, Cython code must, unlike Python, be compiled. This usually happens in two stages:

- a *.pyx* file is compiled by Cython into a *.c* file;
- the *.c* file is then compiled to a *.so* file which can be imported and used directly in any Python session.

The algorithm we implemented uses dynamic programming and has a worst case cost of $\mathcal{O}(n \times m)$ where n and m are the sequences lengths. The main difference between the classical implementation, that works by thoroughly filling a $n \times m$ table and eventually uses backtracking to recover the final result, and our approach, consists in how we handle the table construction. In fact, unlike the

original implementation, we update the table only when a match is found, thus reducing the whole computational cost.

However, for our task, we found out that the number of updates required is, in average, around the 70% of the table size. This is most likely due to the reduced alphabet size (four letters) of the sequences. For this reason the total cost of the function is comparable with the regular implementation.

In average the Cython version of the algorithm is 20 times faster than the original Python implementation and this justifies its use in the DeepMiRNA pipeline.

4.3.3 Filtering predictions

In chapter 3 we highlighted the importance of site accessibility for miRNA targets prediction. In fact, many studies [19] [7] have proved genes accommodate site accessibility by preferentially positioning targets in highly accessible regions [21] thus demonstrating that target accessibility may be a useful feature.

Moreover, the use of a greedier CSSM approach may give rise to an increase on the number of potential candidate sites for each miRNA:mRNA duplex. For example, on average, where a strict canonical approach identifies $\approx$ 3-4 sites per duplex, CSS-6.0:10 identifies approximately 32 MBSs while CSS-7.0:10 about 21 and CSS-7.1:10 around 15. As a consequence the chance of obtaining more false positives is very likely to increase.

This could represent a problem for the neural network, because it may not be able to completely discern false from true positive duplexes. This could be due, in particular, to two main factors: first of all the number of validated negative MBSs is extremely low; in fact, despite having a good number of negatively validates miRNA:mRNA pairs, we only have a small quantity of negatively validated binding sites. Thus, we had to artificially select the negative sites for the training stage as explained in the previous section. Second, functional and non-functional sites are very similar in terms of complementarity with the pairing miRNA and, hence, we believe it's important to also consider the characteristics of the secondary structure of the duplex to improve the accuracy of the prediction and reduce the number of false positives.

To this purpose, we investigated the possibility of using an a-posteriori filter that considers the secondary structure and compute the site accessibility of the target site. The site accessibility has been computed as follows:

- for each potential site identified using the CSSM, we considered the region of 200 nucleotides surrounding the MBS and we call it *folding chunk*. For example, if the MBS has length 30 and comprises the region between nucleotide 100 and 129, we consider for the computation of the site accessibility the region between nucleotide 15 and 214. If either on the left or on the right-hand

side of the MBS there were not enough nucleotides we considered a shorter fold;

- the free energy ΔG_{free} of the folding region is computed, using Vienna Cofold, to check the amount of energy released during the reaction;
- next we computed the opening energy ΔG_{open}, that is the energy needed to unfold the mRNA and allow the miRNA binding;
- the site accessibility energy $\Delta\Delta G$ is given by the difference between the free energy released with the binding and the opening energy (see figure 13). The bigger the value the more accessible the site:

$$\Delta\Delta G = \Delta G_{free} - \Delta G_{open} \tag{1}$$

In order to use this feature as a filter we set a site accessibility threshold meaning that sites with a low value are discarded while only MBS with a site accessibility greater than the threshold are considered as functional.

The use of the filtering stage has proved to be useful in case of more complex networks such as convolutional models, while for simpler network such as the feed-forward model available in DeepMiRNA this feature did not provide any significant benefit.

In addition, one may wonder why we used this feature as an a-posteriori filter rather than using it as another candidate selection rule. While the final result is the same, computing the secondary structure of a folding chunk, and hence its accessibility, is a pretty difficult task involving a lot more computation than asking the network for a prediction. Thus, we decided to use this feature as the last pipeline step.

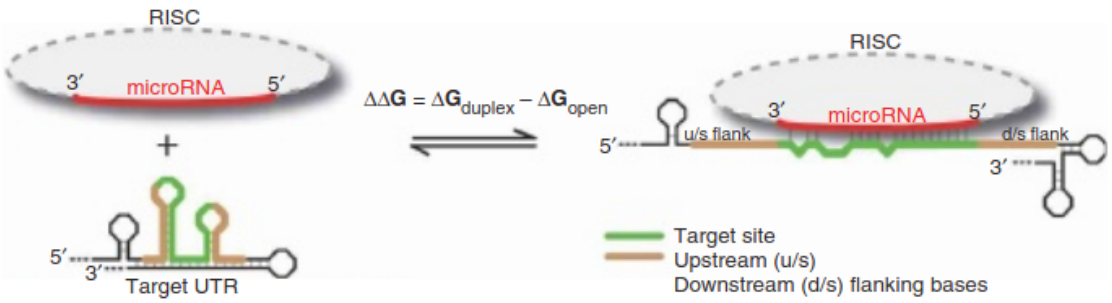

Figure 13: Illustration of site accessibility energy for miRNA:MBS interactions.

4.3.4 Obtaining the final classification

In order to classify a given duplex, we consider that the miRNA targets a gene if any of the potential MBSs of the mRNA are predicted to be functional. The targeting process implemented in DeepMiRNA requires the neural network to classify at least one not filtered candidate site as functional to consider a miRNA:mRNA as a positive targeting event.

More specifically, given a miRNA m, a gene g and a free energy threshold for the duplex th_{duplex}, a candidate site selection method $sm(m, g, th_{duplex})$ determines a set CS of potential MBSs, considering both partial complementarity and stability of the bond.

For this last requirement we chose a threshold $th_{duplex} = -10kcal/mol$, meaning that only miRNA:MBS pairs with a free energy lower than this threshold are kept:

$$sm(m, g, th_{duplex}) = CS \tag{2}$$

To determine if the the miRNA is targeting the gene, each candidate site within the miRNA:mRNA segment is properly encoded and input to the neural network. The result of the targeting prediction $T(m, g)$ corresponds to the disjunction of the conjunction between the neural network outputs for all the candidate sites and the site accessibility check:

$$T(m, g) = \bigvee_{cs} nn(m, cs) \wedge \Delta\Delta G(cs) > th_{sa} \tag{3}$$

Where $nn(m, cs)$ denotes the output of the neural network while $\Delta\Delta G(cs) > th_{sa}$ represents the filtering step checking the site accessibility of the candidate is sufficiently high. When the filter is not used this term is always equal to 1.

The best performance has been achieved by setting $th_{sa} = -14kcal/mol$.

Chapter 5

Neural network design and implementation

5.1 External libraries

DeepMiRNA has been developed using *Python3*. The *requirements.txt* file, present in the project root directory, contains names and versions of all libraries used for the implementation. In particular, we used *Pandas* and *Numpy* for the preprocessing step and the datasets preparation together with *RNA* (the Python version of the Vienna Cofold) and *Biopython* to parse .fasta files and convert them to .csv.

Implementation of the neural network was done with *Keras*[40] using *Tensorflow* backend[41]. Keras is an open-source neural-network library capable of running on top of TensorFlow, Theano and other major deep learning frameworks. Keras contains numerous implementations of commonly used NN building blocks such as layers, optimizers or activation functions and it aims at being user-friendly, modular and extensible. However, Keras, which is now fully supported in the Tensorflow's core library, has been conceived more as an interface than a standalone framework.

We opted for the use of this library because it offers a higher-level, more intuitive set of abstractions that makes it easier to develop deep learning models.

5.2 Choosing the right model

Building deep learning applications in the real world is a never-ending process of selecting and refining the right elements of a specific solution. Among those elements, the selection of the correct model and the right structure of the training dataset are, arguably, the two most important decisions that any data scientist

needs to make when designing deep learning solutions. How to decide what deep learning model to use for a specific problem? How do we know whether we are using the correct training dataset or we should gather more data? Those questions are the common denominator across all stages of the life cycle of a deep learning application. Even though there is no magic answer to those questions, there are several ideas that could guide the decision process.

First of all we need to start identifying the correct baseline model, in particular we should select what type of networks suits more the input dataset. In the case of miRNA targets predictions the topological structure of the available data are strictly correlated to the vectorization method selected. If we opt for the one-hot encoding that maps duplexes into fixed-size vectors, we should be thinking of using a feed-forward network with inter layer connectivity. While, if we select the Dna2Vec approach that transforms each duplex into a matrix, then the problem could be tackled using convolutional neural networks (CNN)[22].

The second part concerns the selection of the optimization algorithm to use. The most popular are, arguably, SGD (Stochastic Gradient Descent) and its variation using momentum or learning decay, and Adam. The latter, in particular, is very often used combined with CNNs.

As mentioned in chapter 4 DeepMiRNA uses 2 different neural network for the training stage according to the chosen data representation: a regular feed-forward network for the one-hot encoded sequences and a CNN for the Dna2Vec encoded duplexes.

5.3 The feed-forward network

The feed-forward network we designed for the miRNA target prediction task is a very simple neural network consisting of 5 dense hidden layers, comprising rectifier activation function ReLU nodes, each followed by a dropout layer. Dropout [42] is a regularization technique used to prevent overfitting. In this thesis, the dropout rate has been set to 0.7 meaning that there is a probability $p = 0.3$ that a certain neuron is ignored (i.e set to 0). Basically this implies that their contribution to the activation of downstream (i.e. next layer) neurons is temporally removed on the forward pass and weight update is not applied on the backward pass (see figure 14).

While a neural network learns, neuron weights settle into their context within the network. Those weights are tuned for specific features providing some specialization. However, neighboring neurons become to rely on this specialization, which, if taken too far, can result in a fragile model too specialized on the training data. Hence, if neurons are randomly dropped out of the network during training,

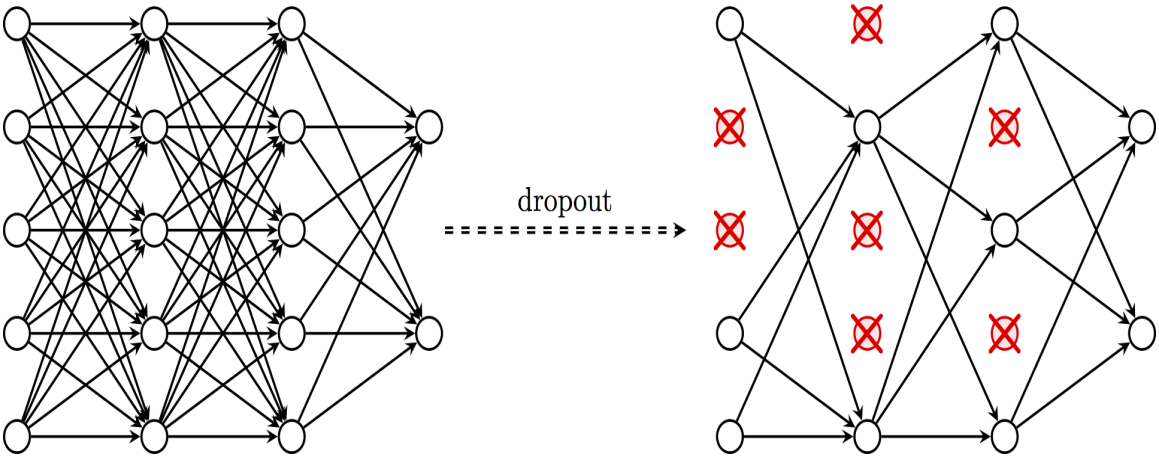

Figure 14: Left: neural network before dropout. Right: neural network after dropout.

other neurons will have to step in and handle the representation required to make predictions for the missing neurons. This is believed to result in multiple independent internal representations being learned by the network. The effect is that the network becomes less sensitive to the specific weights of neurons and results in a model capable of better generalization and less likely to overfit the training data.

The output layer is composed of one sigmoid node that returns a value between 0 and 1 corresponding to the final score prediction. The class of the site is determined by the output of this neuron:

$$class(i) = \begin{cases} 1 & \text{if } o \geq 0.4 \\ 0 & \text{if } o < 0.4 \end{cases}$$

Where i and o denote respectively the input and the output of the network. The prediction threshold value 0.4 has been empirically evaluated.

For the loss function we chose the binary cross entropy with Adam optimizer set with the following parameters:

- learning rate = 0.002;
- beta_1 = 0.9;
- beta_2 = 0.999;
- epsilon = $1e-8$;
- decay = 0.0

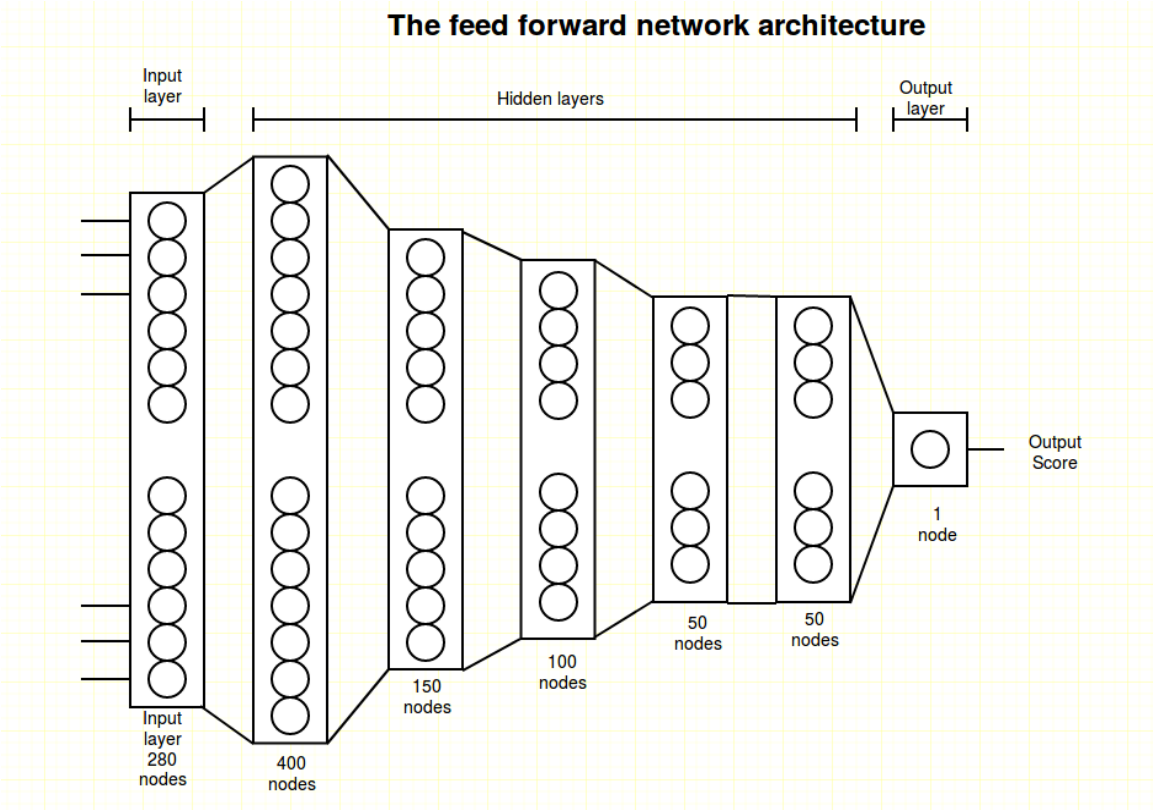

Figure 15: **The feed-forward neural network**. The first hidden layer increases the dimensionality of the input allowing the representation of data in a more complex feature space. There is debate in the machine learning community regarding the need of such over-completion layers, as they do not necessarily improve the accuracy whilst making the learning process slower. Considering that the relatively low number of inputs of the proposed network allows a fast training procedure, we opted to include this higher dimension layer to give the network the chance of identifying more complex patterns.

The resulting architecture can be visualized in figure15 and parameters such as number of layers and neurons have been chosen through cross validation.

Despite being a very simple model, this network showed a better performances on the test set compared to other deeper (that is with more hidden layers) or more complex (i.e with a greater number of nodes) models. Check chapter 6 for further details on the results.

5.4 The Convolutional Neural Network aka CNN

We have explained earlier, that the second encoding function provided by DeepMiRNA is based on Dna2Vec [43]. In computational biology, one of the most widely used representation of long DNA sequence consists in dividing it into shorter k-mer components. Unfortunately, the straightforward vectorization of k-mer as a one-hot vector is vulnerable to the curse of dimensionality. Worse yet, the distance between any pair of one-hot vectors is equidistant, even though, for example, the sequence *ATGGC* should be closer to *ATGGG* than *CACGA*. This might be particularly problematic when applying the latest machine learning algorithms to solve problems in biological sequence analysis. Dna2Vec is, thus, a distributed representation of biological sequences that allows mapping k-mers to dense vectors of real numbers. By splitting the miRNA:MBS duplex in a fixed number of variable length k-mers, we were able to encode each pair into a bi-dimensional matrix.

More specifically, each k-mer is embedded into a continuous vector space of 100 dimensions, and each duplex is divided into 9 variable length k-mers: concatenating these vectors we obtain a $9x100$ matrix representing the encoded sequence (see figure 16).

With this representation, we believe the best base model to use for our task is represented by a convolutional neural network.

5.4.1 CNN architecture

Once the baseline model has been chosen, one must proceed with its implementation selecting the best possible design. This step can be very tricky, especially when dealing with convolutional networks. In fact, according to the complexity of the architecture and the representation of the available data, a CNN may require a long training time, with a great number of epochs. This makes the evaluation of the model design a complicated task: large validation errors and little improvements in the first few epochs does not always imply that the chosen architecture is wrong.

For DeepMiRNA, the network was configured so that the number of inputs in the input layer was equal to the dimensionality of the encoded sequences, while the output layer consisted of two softmax neurons.

The architecture of our convolutional network is shown in figure (add figure here) and it is mainly inspired to the CNN used in [44]. Given that our model must be trained on a relative small dataset we decided to adopt a single level of convolutional feature map followed by a non-linear ReLU activation layer and a max pooling layer immediately preceding the output softmax neurons.

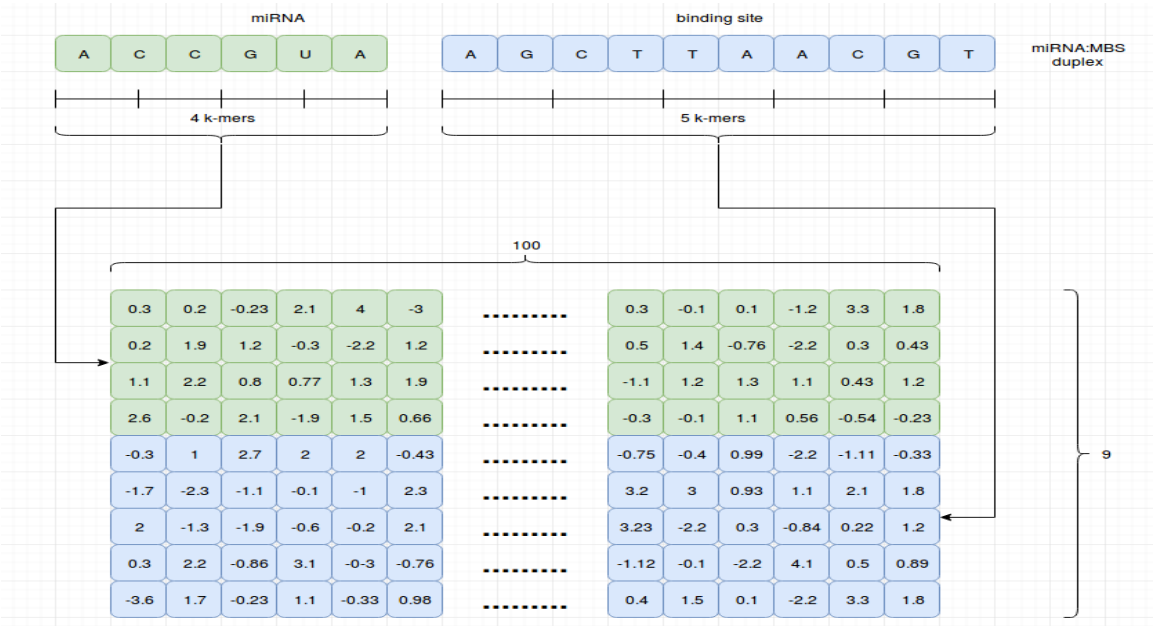

Figure 16: **Sequence Embedding.** In green the miRNA sequence is split into 4 k-mers each mapped in a 100-dimension vector. In blue the same is done with miRNA binding site which is divided into 5 k-mers. The concatenation of the 9 resulting vectors is the matrix distributed representation of the duplex.

For the compiling step required for the training stage, we selected the binary cross entropy loss function and we set the Adam optimizer with the same parameters used for the MLP network.

Despite showing discrete results in terms of accuracy and F1-score on the validation set composed of never seen miRNA:MBS examples, the CNN does not perform very well on the test set especially if compared to the feed-forward network. The main issue concerns its scarce ability in identifying correctly negative miRNA:mRNA pairs that prevents this model from having a good generalization capability. We believe the main reason behind this, it's most likely due to the very small amount of negatively validated entries in the training set: issue that, instead, does not seem to affect the MLP model performance.

Chapter 6

Experimental results

6.1 Introduction

In the previous chapters we described how the DeepMiRNA's architecture is based on two main functional blocks: on one side there is the NN, whose purpose is that of analyzing the selected candidate target sites, while on the other there are the CSSM used during the target prediction step and the a-posteriori filter that tries to refine the network prediction.

To assess these two aspects, we first evaluated the outputs of the NN training process through cross validation and then investigated performance using the different candidate site selection methods outlined in chapter 5. These comprised the novel (non-canonical) models implemented for this thesis: CSS-6.10 and CSS-7.10. Finally, we tested DeepMiRNA's performance by comparing it against TargetScan[35], PITA[21] and miRAW [38], which represent the most commonly used target site predictors based on citations.

The experimental results present in this chapter concern the best pipeline configuration for DeepMiRNA. This comprises the use of the feed-forward neural network that exhibited an overall better performance than the convolutional model, especially in the identification of negative binding sites.

There can be many reason explaining this differences, but we believe that the lack of a consistent dataset of negatively validated binding sites may be the principal cause.

6.2 Neural Netework evaluation

The best performing feed-forward neural network was trained using the whole training set with this parameters:

- dropout rate = 0.7
- optimizer = Adam as described in chapter 5
- loss function = binary cross entropy
- batch size = 128
- number of epochs = 15

The above parameters where computed using a validation set composed of about 10000 examples (20% of the training set).

Once the best model has been selected, we built a fresh network using the best parameters and we trained it holding out the 25% of the training data for the purpose of testing its performance. This validation presented very good results in terms of predicting both positive and negative sites, with all evaluated metrics resulting in scores well above 0.9 (see figure17).

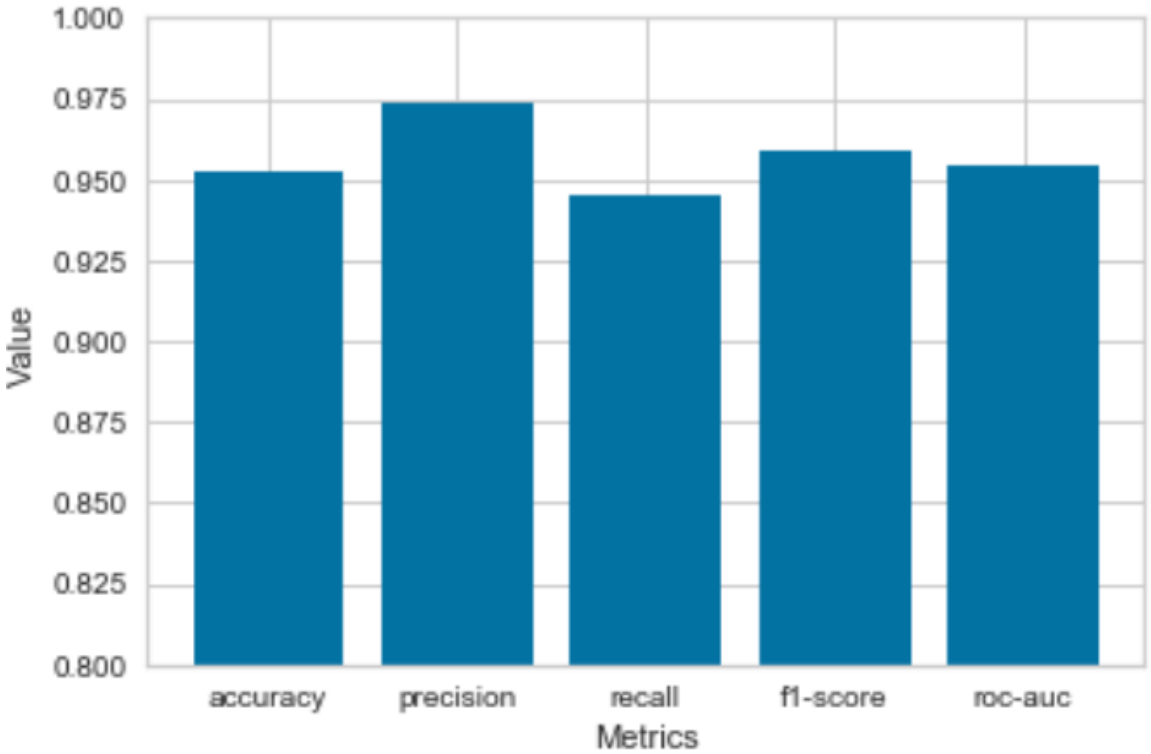

Figure 17: **Network performance metrics.** To this purpose we trained the network with best parameters using 75% of the training data. The remaining 25% has been used as test set. As shown above all metrics are close to 0.95.

6.3 The role of the candidate site selection method

The purpose of using a CSSM is to narrow the search space to simplify the neural network classification task. To investigate the impact of the site selection method, we compared the performances of three different methods as described in chapter4.

Figure 18 summaries the obtained results: all methods reach accuracies between 0.8 and 0.82, these values are computed considering the imbalance of the test set, that is they are calculated as the average of the accuracies on positive and negative samples. CSS-6.0:10 seems to achieve slightly better performances for every metric but precision, compared to the other two methods. The F1-score is the harmonic average between precision and recall and shows how well both classes are classified by a particular CSSM. The values obtained are very similar to the accuracies and indicate an ability to effectively predict both negative and positive targets especially for CSS-6.0:10 and CSS-7.0:10.

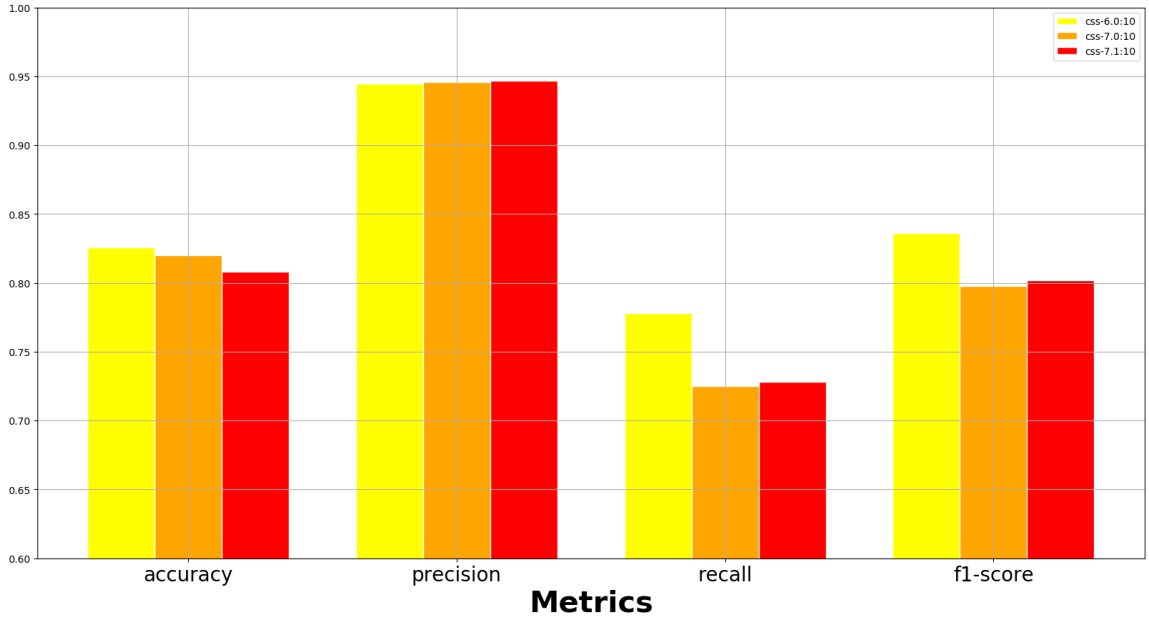

Figure 18: **Evaluation of DeepMiRNA different candidate site selection methods.** Results are evaluated in terms of balanced accuracy, precision, recall and f1-score. The best result was achieved using CSS-6.0:10.

Candidates selection also involved the choice of a threshold for the binding energy th_{duplex}. The idea was to only retain duplexes with a low free energy and, hence, able to form a stable bond. In order to set this value, we randomly extracted 10 non-overlapping folds from the test set, each composed of 5000 entries, and we

averaged the balanced accuracies obtained by each possible choice. Table 3 shows that the best result has been obtained setting $th_{duplex} = -10$ kcal/mol.

Table 3: **Free energy threshold evalution.**

Threshold	Balanced Accuracy
0	0.79
-2	0.792
-4	0.798
-6	0.824
-8	0.849
-9	0.857
-10	**0.861**
-11	0.85
-12	0.835
-14	0.788
-16	0.74

Higher values produce longer lists of candidate sites and potentially increase the number of false positives, while low values may filter functional binding sites and produce a higher number of false negatives.

6.4 Site accessibility filter

In order to measure the effect of site accessibility on miRNA target prediction, we tested the best performing pipeline configuration without filtering the network output. While this reduces the computational cost of the whole process, DeepMiRNA becomes slightly more biased towards the prediction of positive sites. However, this is true only for the convolutional model. To underline this difference we report the results obtained with both network models.

Table 4 shows that using a site accessibility filter (WF) with the feed-forward neural network does not improve the prediction performance, in fact, a no filter (NF) configuration actually achieves better results. This means that the rate of false negatives generated by the filter is higher than the gain derived from the reduction of the false positives. Achieving high precision means that when the network output a 1 is almost never fails, while high values of recall implies the number of false negative is low. These results show that the feed-forward network is very precise in identifying negative binding sites, while it behaves worse with positive targets.

For the CNN, instead, the filtering steps improves the network ability to predict miRNA's targets correctly. Table 4 indicates that precision, which is the measure of how confident the system is predicting positive values, and balanced accuracies increase, while recall, that is the fraction of positive instances that have been correctly classified over the total amount of validated positive entries, decreases much slower.

Table 4: **Results with or without the a-posteriori filter.**

Classifier	Accuracies		Precision		Recall		F1-score	
	NF	*WF*	*NF*	*WF*	*NF*	*WF*	*NF*	*WF*
Feed-forward	**0.827**	0.82	0.936	**0.939**	**0.794**	0.785	**0.840**	0.838
CNN	0.628	**0.652**	0.649	**0.761**	**0.662**	0.621	0.64	**0.681**

6.5 Comparison with other miRNA target prediction tools

Figure 19 compares the performance of DeepMiRNA best configuration, that is CSS-6.0:10 without filter, with state of art target prediction software tools TargetScan, PITA and miRAW using the test set defined in chapter 4.

For TargetScan we exploited both available configurations: TS conserved which uses an additional feature about interspecies conservation of miRNA and TS non-conserved which does not. For both approaches, the low accuracies seem a consequence of their tendency to misclassify true targets as negative; i.e., despite reporting high precision (> 0.9), their recall, and hence their F1-score, was low and so was their balanced accuracies.

Conversely, PITA reported better accuracy and F1-score, but still lower than the one achieved by miRAW and DeepMiRNA. According to precision, recall and, hence, F1-score, miRAW perform slighty better than our tool, however the accuracy it achieves is very low (0.55). Balanced accuracy is the average between the

Table 5: **TPR and TNR comparison between miRAW and DeepMiRNA.**

NP	NN	TP	FP	TN	FN	TPR	TNR	BA
195825	11100	84%	75%	25%	16%	0.84	0.254	0.547
195825	11100	80%	17%	83%	20%	0.805	0.833	0.82

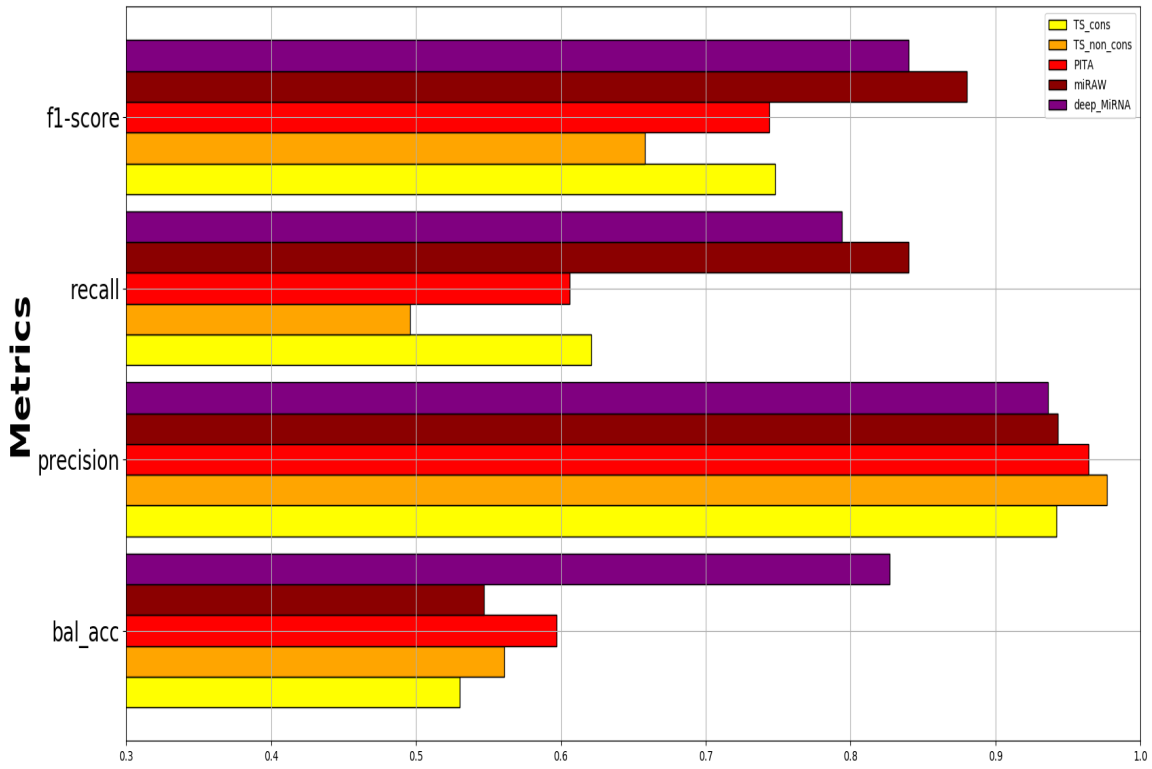

Figure 19: **Comparison of DeepMiRNA with other commonly used prediction tools.** Even though miRAW performs better than DeepMiRNA in most of the metrics, it only has 0.55 balanced accuracy. This is caused by its poor ability of identifying negative interactions. TargetScan (TS), in both its configurations, and PITA adopt a too conservative approach that inhibits their capacity of recognizing positive duplexes correctly.

true positive (TPR) and the true negative rate (TNR); to explain why miRAW reported such a low value for this metric we can check Table 5.

NP and NN identify respectively the number of positive and negative validated pairs in the test set, while TP, FP, TN and FN indicate the percentage of true/false positive/negative predictions. In particular, these values have been computed as the fraction of predicted values of a certain class over the total size of the same class. So for example a TP value of 84% states that of all positive samples of the test set 84% were classified correctly, while a FN of 16% indicates that for the negative class 16 samples out of 100 were misclassified. Those values help in giving a feel of what TPR and TNR represent. Finally, BA stands for balance accuracy.

MiRAW has a true positive rate of 0.84 indicating its better ability in identifying positives miRNA:mRNA pairs compared to the other tools. However, a large

number of negative sites has also been misclassified as positive (75%), meaning that it performed poorly on negative entries. This is confirmed by a very low true negative rate which gives rise to a lower value of balanced accuracy.

Conversely, DeepMiRNA shows a good ability in predicting both classes, even though it tends to be slightly biased towards negative predictions.

This tendency is reflected in the confusion matrix shown in figure 20 concerning DeepMiRNA's performance over the complete test set.

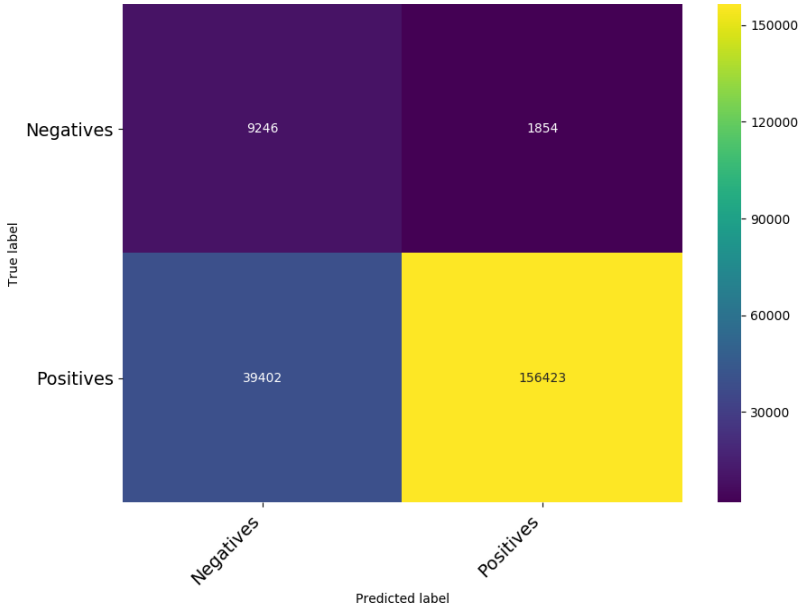

Figure 20: **DeepMiRNA Confusion matrix for the test set.** This matrix shows that DeepMiRNA performs well on both classes.

Chapter 7

Conclusions

7.1 Final considerations

MiRNA's targeting is a complex, yet not fully understood, mechanism. In an ideal scenario it would be possible to experimentally verify the target set for all miRNAs, but both the cost and the limited throughput of current methods imply that miRNA studies still depend on computational predictions to complement experimental data.

On the other side, its imprecise nature makes the decision of which computational approach to employ for the task, extremely challenging. Often, one of the hardest part of solving any Machine Learning problem, can be, in fact, finding the right algorithm for the job. A certain model is better suited for different types of data and different problems and it's never easy to pick the right one. This is also due to the so called 'no free lunch theorem' [45], which roughly states that there is no 'perfect' Machine Learning algorithm that will perform well at any problem.

For the miRNA's targets prediction task, however, the need of finding suitable patterns to distinguish functional binding sites from non-functional, guided our choice towards a neural network.

Nowadays, deep learning has been exploited in many fields for various tasks, from image recognition to natural language processing with different results. The main advantage of its use consists in its capacity to automatically extract its own data feature descriptors. However, the more complex the network architecture, the more computational power (and data) it needs to properly extract valuable knowledge.

In this thesis we tackled the problem of miRNA target classification adopting a neutral approach towards the prediction process, avoiding incorporating any hand-crafted feature related to the targeting process. The obtained results and the comparison between DeepMiRNA and other descriptor-based tools suggest

that current knowledge is still not sufficient to accurately capture all aspects of the miRNA targeting process. In fact DeepMiRNA's better performance over most of the available feature-based predictors, indicates that the descriptors learned by our neural network are able to encode both current knowledge and additional information yet to be determined.

Undoubtedly, the most challenging part in DeepMiRNA's development was the processing of different data sources to build representative train, test and validation sets. For this work we selected Diana TarBase and mirTarBase as our core data sources because they represented the most comprehensive set of evidence for miRNA:mRNA functional interactions. However, for most of the validated experiments the databases do not provide exact details of the target site for the supported interactions and do not contain data about the exact transcript of the experiment but only the identification of the gene. Unfortunately, any gene is associated with a variable number of transcripts and this lack of information forced us to make the decision of selecting the one with the longest sequence, potentially inserting unwanted noise to the data. Also, in order to obtain reliable binding site information we integrated PAR-CLIP [12] and CLASH [19] datasets, which provided us with these information, and cross-referenced them with the main datasets. These additional data, though, turned out to be almost exclusively related to experimentally verified positive data, containing only a few examples of negative validates pairs. This, again, lead us to the decision of artificially creating negative samples to be used for training stage.

Thus, we can affirm that the amount of validated data available for the task is still not sufficient to capture all characteristics of the targeting process; this is the reason why we incorporated a set of rules to select the best binding sites candidates to be processed by the neural network. In an ideal scenario, with enough representative positive and negative data samples, this step could be skipped as a deep enough neural network should be able to map such information into its weights. Moreover, the requirement for partial complementarity within the seed region defining the CSSM used, seems to be universally accepted and established through numerous experimental studies [7].

The binding sites selection steps also involved the relaxation of the canonical seed region into an extended miRNA subsequence comprising 10 nucleotides from the first to the tenth (in the best configuration), to accommodate the presence of non-canonical targets: we called this sequence *extended seed region*. This choice resulted in a better overall performance compared to other softwares utilizing a more conservative approach that considers only canonical binding sites and suggests that perfect seed region complementarity is not a sufficient discriminant to correctly identify miRNA's targets.

Another important task within this thesis was the choice of the neural network

structure. Inspired by this article [46] we decided to face the vectorization step using two different algorithms: on one side the classical one-hot encoding of the sequences and on the other the continuous representation of biological sequence based on Dnad2Vec [43]. This gave us the opportunity to tackle the miRNA targeting process from two different points of views.

Testing results have decreed that the classical approach using a MLP design offers a more reliable and robust classifier. This is most likely due to the lack of complete representativeness exhibited by the available data and the more complex structure of a CNN that requires a wider range of training data to correctly set its weights.

7.2 Future work

Despite the enhanced performance demonstrated by DeepMiRNA, it is prudent to consider some of the potential limitations of automatic feature learning approaches such as DL. The hierarchical internal data representation learned by a neural network can be sometimes be difficult to interpret and map into human interpretable knowledge, hence it is not possible to directly identify the features that determine the classification. We tried to address this issue by looking at the weights learned after the training process using specific techniques such as Grad-CAM [47] and other visualization methods such as the ones described in this article [48], but unfortunately we were not able to extract any valuable information. It's worth mentioning, though, that further investigations in that direction may aid the interpretation process and improve the classification accuracy, which is the next logical step in our work.

Another important improvement to DeepMiRNA could concern the extension to the whole gene transcript rather than only analyzing the 3' untranslated region. Recent studies [49] [19], in fact, revealed the importance of the whole sequence, comprising both coding and 5' untranslated regions, in miRNA's targets prediction. Although over 60% of miRNA:mRNA interactions take place inside the 3'UTR, we believe that extending the search of potential candidate sites to other regions is most likely to improve the accuracy of the predictor.

For the development of DeepMiRNA, anyhow, we decided to only use data relative to the 3'UTR because it has still not been fully clarified if actual bindings to other regions exhibit the same potential to repress mRNA translation.

Besides, even though the work presented in this thesis focused on the prediction of human miRNA targets, the same methodology can be applied to build target prediction models for any other living species, aware that gathering enough representative data will be of crucial importance for the new task.

We conclude this thesis by considering that the presented approach will certainly benefit from further experimental studies that will serve to validate new predictions obtained by DeepMiRNA, but also to generate new experimental data to reliably expand the training of the model in both its configurations.

Appendix A

Frequently Asked Questions

A.1 How do I change the colors of links?

The color of links can be changed to your liking using:

`\hypersetup{urlcolor=red}`, or

`\hypersetup{citecolor=green}`, or

`\hypersetup{allcolor=blue}`.

If you want to completely hide the links, you can use:

`\hypersetup{allcolors=.}`, or even better:

`\hypersetup{hidelinks}`.

If you want to have obvious links in the PDF but not the printed text, use:

`\hypersetup{colorlinks=false}`.

Bibliography

[1] David Barthel. ibiology: Introduction to mirnas. https://www.ibiology.org/genetics-and-gene-regulation/introduction-to-micrornas/. Accessed: 2019-06-11.

[2] Lee P Lim, Nelson C Lau, Philip Garrett-Engele, Andrew Grimson, Janell M Schelter, John Castle, David P Bartel, Peter S Linsley, and Jason M Johnson. Microarray analysis shows that some micrornas downregulate large numbers of target mrnas. *Nature*, 433(7027):769, 2005.

[3] Benjamin P Lewis, Christopher B Burge, and David P Bartel. Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microrna targets. *cell*, 120(1):15–20, 2005.

[4] Bastian Fromm, Tyler Billipp, Liam E Peck, Morten Johansen, James E Tarver, Benjamin L King, James M Newcomb, Lorenzo F Sempere, Kjersti Flatmark, Eivind Hovig, et al. A uniform system for the annotation of vertebrate microrna genes and the evolution of the human micrornaome. *Annual review of genetics*, 49:213–242, 2015.

[5] Yuka Watanabe, Masaru Tomita, and Akio Kanai. Computational methods for microrna target prediction. *Methods in enzymology*, 427:65–86, 2007.

[6] Ming Lu, Qipeng Zhang, Min Deng, Jing Miao, Yanhong Guo, Wei Gao, and Qinghua Cui. An analysis of human microrna and disease associations. *PloS one*, 3(10):e3420, 2008.

[7] Sarah M Peterson, Jeffrey A Thompson, Melanie L Ufkin, Pradeep Sathyanarayana, Lucy Liaw, and Clare Bates Congdon. Common features of microrna target prediction tools. *Frontiers in genetics*, 5:23, 2014.

[8] Ray M Marín and Jiří Vaníček. Efficient use of accessibility in microrna target prediction. *Nucleic acids research*, 39(1):19–29, 2010.

[9] Amy E Pasquinelli. Microrna and their targets: recognition, regulation and an emerging reciprocal relationship. *Nature Reviews Genetics*, 13(4):271, 2012.

[10] Vikram Agarwal, George W Bell, Jin-Wu Nam, and David P Bartel. Predicting effective microrna target sites in mammalian mrnas. *elife*, 4:e05005, 2015.

[11] Pierre Maziere and Anton J Enright. Prediction of microrna targets. *Drug discovery today*, 12(11-12):452–458, 2007.

[12] Stefanie Grosswendt, Andrei Filipchyk, Mark Manzano, and Klironomos. Unambiguous identification of mirna: target site interactions by different types of ligation reactions. *Molecular cell*, 54(6):1042–1054, 2014.

[13] Martin Reczko, Manolis Maragkakis, Panagiotis Alexiou, Ivo Grosse, and Artemis G Hatzigeorgiou. Functional microrna targets in protein coding sequences. *Bioinformatics*, 28(6):771–776, 2012.

[14] J Robin Lytle, Therese A Yario, and Joan A Steitz. Target mrnas are repressed as efficiently by microrna-binding sites in the 5'utr as in the 3'utr. *Proceedings of the National Academy of Sciences*, 104(23):9667–9672, 2007.

[15] Doyeon Kim, You Me Sung, Jinman Park, Sukjun Kim, Jongkyu Kim, Junhee Park, Haeok Ha, Jung Yoon Bae, SoHui Kim, and Daehyun Baek. General rules for functional microrna targeting. *Nature genetics*, 48(12):1517, 2016.

[16] Sinéad M Smith and David W Murray. An overview of microrna methods: expression profiling and target identification. In *Molecular Profiling*, pages 119–138. Springer, 2012.

[17] Nicole T Schirle, Jessica Sheu-Gruttadauria, and Ian J MacRae. Structural basis for microrna targeting. *Science*, 346(6209):608–613, 2014.

[18] Tingting Du and Phillip D Zamore. microprimer: the biogenesis and function of microrna. *Development*, 132(21):4645–4652, 2005.

[19] Aleksandra Helwak, Grzegorz Kudla, Tatiana Dudnakova, and David Tollervey. Mapping the human mirna interactome by clash reveals frequent noncanonical binding. *Cell*, 153(3):654–665, 2013.

[20] Dong Yue, Hui Liu, and Yufei Huang. Survey of computational algorithms for microrna target prediction. *Current genomics*, 10(7):478–492, 2009.

[21] Michael Kertesz, Nicola Iovino, Ulrich Unnerstall, Ulrike Gaul, and Eran Segal. The role of site accessibility in microrna target recognition. *Nature genetics*, 39(10):1278, 2007.

[22] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[23] Shuang Cheng, Maozu Guo, Chunyu Wang, Xiaoyan Liu, Yang Liu, and Xuejian Wu. Mirtdl: a deep learning approach for mirna target prediction. *IEEE/ACM transactions on computational biology and bioinformatics*, 13(6):1161–1169, 2015.

[24] Byunghan Lee, Junghwan Baek, Seunghyun Park, and Sungroh Yoon. deeptarget: end-to-end learning framework for microrna target prediction using deep recurrent neural networks. In *Proceedings of the 7th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 434–442. ACM, 2016.

[25] Ming Wen, Peisheng Cong, Zhimin Zhang, Hongmei Lu, and Tonghua Li. Deepmirtar: a deep-learning approach for predicting human mirna targets. *Bioinformatics*, 34(22):3781–3787, 2018.

[26] Ivo L Hofacker. Vienna rna secondary structure server. *Nucleic acids research*, 31(13):3429–3431, 2003.

[27] Onehot encoding in python. `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.OneHotEncoder.html`. Accessed: 2019-06-15.

[28] Yoav Goldberg and Omer Levy. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[29] Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas, et al. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, 30(1):25–36, 2006.

[30] Ioannis S Vlachos, Maria D Paraskevopoulou, Dimitra Karagkouni, Georgios Georgakilas, Thanasis Vergoulis, Ilias Kanellos, Ioannis-Laertis Anastasopoulos, Sofia Maniou, Konstantina Karathanou, Despina Kalfakakou, et al. Diana-tarbase v7. 0: indexing more than half a million experimentally supported mirna: mrna interactions. *Nucleic acids research*, 43(D1):D153–D159, 2014.

[31] Sheng-Da Hsu, Feng-Mao Lin, Wei-Yun Wu, Chao Liang, Wei-Chih Huang, Wen-Ling Chan, Wen-Ting Tsai, Goun-Zhou Chen, Chia-Jung Lee, Chih-Min Chiu, et al. mirtarbase: a database curates experimentally validated microrna–target interactions. *Nucleic acids research*, 39(suppl_1):D163–D169, 2010.

[32] Andrew Yates, Wasiu Akanni, M Ridwan Amode, Barrell, et al. Ensembl 2016. *Nucleic acids research*, 44(D1):D710–D716, 2015.

[33] Markus Hafner, Markus Landthaler, Lukas Burger, Mohsen Khorshid, Jean Hausser, Philipp Berninger, Andrea Rothballer, Manuel Ascano Jr, Anna-Carina Jungkamp, Mathias Munschauer, et al. Transcriptome-wide identification of rna-binding protein and microrna target sites by par-clip. *Cell*, 141(1):129–141, 2010.

[34] Grzegorz Kudla, Sander Granneman, Daniela Hahn, Jean D Beggs, and David Tollervey. Cross-linking, ligation, and sequencing of hybrids reveals rna–rna interactions in yeast. *Proceedings of the National Academy of Sciences*, 108(24):10010–10015, 2011.

[35] Vikram Agarwal, George W Bell, Jin-Wu Nam, and David P Bartel. Predicting effective microrna target sites in mammalian mrnas. *elife*, 4:e05005, 2015.

[36] Mark Menor, Travers Ching, Xun Zhu, David Garmire, and Lana X Garmire. mirmark: a site-level and utr-level classifier for mirna target prediction. *Genome biology*, 15(10):500, 2014.

[37] Tara Boyle. Dealing with imbalanced data, 2019. Accessed: 2019-06-25.

[38] Albert Pla, Xiangfu Zhong, and Simon Rayner. miraw: A deep learning-based approach to predict microrna targets by analyzing whole microrna transcripts. *PLoS computational biology*, 14(7):e1006185, 2018.

[39] S. Behnel, R. Bradshaw, C. Citro, L. Dalcin, D.S. Seljebotn, and K. Smith. Cython: The best of both worlds. *Computing in Science Engineering*, 13(2):31 –39, 2011.

[40] François Chollet et al. Keras. `https://keras.io`, 2015.

[41] Martin et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[42] Nitish Srivastava, Geoffrey Hinton, and Ruslan Krizhevsky. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[43] Dhananjay Kimothi, Akshay Soni, Pravesh Biyani, and James M Hogan. Distributed representations for biological sequence analysis. *arXiv preprint arXiv:1608.05949*, 2016.

[44] Aliaksei Severyn and Alessandro Moschitti. Unitn: Training deep convolutional neural network for twitter sentiment classification. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 464–469, 2015.

[45] David H Wolpert, William G Macready, et al. No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, 1(1):67–82, 1997.

[46] Ehsaneddin Asgari and Mohammad RK Mofrad. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PloS one*, 10(11):e0141287, 2015.

[47] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

[48] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*, 2015.

[49] Harsh Dweep, Carsten Sticht, Priyanka Pandey, and Norbert Gretz. mirwalk–database: prediction of possible mirna binding sites by "walking" the genes of three genomes. *Journal of biomedical informatics*, 44(5):839–847, 2011.