deepTarget: End-to-end Learning Framework for microRNA Target Prediction using Deep Recurrent Neural Networks

Byunghan Lee Electrical and Computer Eng. Seoul National University Seoul 08826, Korea

Seunghyun Park
Electrical and Computer Eng.
Seoul National University
Seoul 08826, Korea
Electrical Engineering
Korea University
Seoul 02841, Korea

Junghwan Baek Interdisciplinary Program in Bioinformatics Seoul National University Seoul 08826, Korea

Sungroh Yoon Electrical and Computer Eng. & Interdisciplinary Program in Bioinformatics
Seoul National University Seoul 08826, Korea sryoon@snu.ac.kr

ABSTRACT

MicroRNAs (miRNAs) are short sequences of ribonucleic acids that control the expression of target messenger RNAs (mRNAs) by binding them. Robust prediction of miRNA-mRNA pairs is of utmost importance in deciphering gene regulation but has been challenging because of high false positive rates, despite a deluge of computational tools that normally require laborious manual feature extraction. This paper presents an end-to-end machine learning framework for miRNA target prediction. Leveraged by deep recurrent neural networks-based auto-encoding and sequence-sequence interaction learning, our approach not only delivers an unprecedented level of accuracy but also eliminates the need for manual feature extraction. The performance gap between the proposed method and existing alternatives is substantial (over 25% increase in F-measure), and deepTarget delivers a quantum leap in the longstanding challenge of robust miRNA target prediction. [availability: http://data.snu.ac.kr/pub/deepTarget]

Categories and Subject Descriptors

I.2.6 [**Learning**]: Connectionism and Neural Nets; I.5.1 [**Models**]: Neural Nets; I.5.2 [**Design Methodology**]: Classifier Design and Evaluation; J.3 [**Life and Medical Sciences**]: Biology and Genetics

General Terms

Algorithms

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

BCB '16, October 02-05, 2016, Seattle, WA, USA © 2016 ACM. ISBN 978-1-4503-4225-4/16/10...\$15.00 DOI: http://dx.doi.org/10.1145/2975167.2975212

Keywords

microRNA, deep learning, recurrent neural networks, LSTM

1. INTRODUCTION

MicroRNAs (miRNAs) are small non-coding RNA molecules that can control the function of their target messenger RNAs (mRNAs) by down-regulating the expression of the targets [4]. By controlling the gene expression at the RNA level, miRNAs are known to be involved in various biological processes and diseases [27]. As miRNAs play a central role in the post-transcriptional regulation of more than 60% of protein coding genes [11], investigating miRNAs is of utmost importance in many disciplines of life science. As explained further in Section 2.3, miRNAs are derived from the precursor miRNAs (premiRNAs) and then exhibit their regulatory function by binding to the target sites present in mRNAs. Two types of computational problems about miRNAs thus naturally arise in bioinformatics: miRNA host identification (*i.e.*, the problem of locating the genes that encode

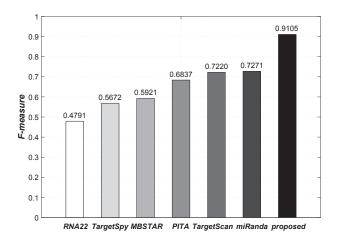


Figure 1: F-measure comparison of miRNA target prediction tools: The proposed deepTarget method significantly outperforms the alternatives (see Section 4 for details)

^{*}To whom correspondence should be addressed.

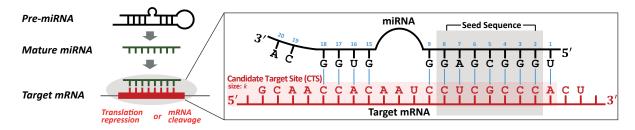


Figure 2: miRNA biogenesis and its function: microRNAs (miRNAs) exhibit their regulatory function by binding to the target sites present in the 3' untranslated region (UTR) of their cognate mRNAs. The *seed* sequence of a miRNA is defined as the first two to eight nucleotides starting from the 5' to the 3' UTR. The *candidate target site* (CTS) refers to the small segment of length k of mRNA. The blue line between a pair of nucleotides (one from miRNA and the other from mRNA) represents the Watson-Crick (WC) pairing formed by sequence complementarity (*i.e.*, the binding of A-U and that of C-G)

pre-miRNAs) and miRNA target prediction (*i.e.*, the task of finding the mRNA targets of given miRNAs). This paper focuses on the target prediction problem.

Due to the importance, there has been a deluge of computational algorithms proposed to address the miRNA target prediction problem [32, 37, 50]. However, most of the existing approaches suffer from two major limitations, often failing to deliver satisfactory performance in practice (see Figure 1). First, the conventional approaches make predictions based on the manually crafted features of known miRNA-mRNA pairs [e.g., the level of sequence complementarity (see Section 2.3) between a miRNA sequence and the sequence of a binding site in its target mRNA]. Manual feature extraction is time-consuming, laborious, and error-prone, giving no guarantee for generality. Second, existing tools often fail to effectively filter out false positives [i.e., bogus miRNA-mRNA pairs that do not actually interact in vivo], producing an unacceptable low level of specificity. This challenge originates from the fact that there are only four types of letters (A, C, G, and U) in RNA sequences and that the length of a typical miRNA sequence is short (about 22 nucleotides), producing a high chance of seeing a random site (in mRNA) whose sequence is complementary to the miRNA.

To boost the sensitivity of miRNA target prediction, a variety of features have been proposed. According to Menor et al. [31], as many as 151 kinds of features appear in the literature, which can be broadly grouped into four common types [37]: the degree of Watson-Crick matches of a seed sequence (see Figure 2); the degree of sequence conservation across species; Gibbs free energy, which measures the stability of the binding of a miRNA-mRNA pair, and the site accessibility, which measures the hybridization possibility of a pair from their secondary structures.

To address the limitations of existing approaches, this paper proposes deepTarget, an end-to-end machine learning framework for robust miRNA target prediction. deepTarget adopts deep recurrent neural network (RNN)-based auto-encoders to discover the inherent representations of miRNA and mRNA sequences and utilizes stacked RNNs to learn the sequence-to-sequence interactions underlying miRNA-mRNA pairs. Leveraged by this combination of unsupervised and supervised learning approaches, deepTarget not only delivers an unprecedented level of accuracy but also eliminates the need for manual feature extraction. Furthermore, according to our visual inspection of the activation of the RNN layers used in deepTarget, meaningful patterns appeared in the nucleotide positions that corresponded to the real miRNA-mRNA binding sites. Further analyzing the activation patterns may allow us to obtain novel biological insight into regulatory interactions.

As shown in Figure 1, the performance gap between deepTarget and

the compared alternatives is substantial (over 25% increase in F-measure), and deepTarget delivers a quantum leap in the long-standing challenge of robust miRNA target prediction.

2. BACKGROUND

As mentioned in the previous section, miRNAs play a crucial role in controlling the function of their target genes by down-regulating gene expression, actively participating in various biological processes. To predict targets of given miRNAs, numerous computational tools have been proposed [37, 45].

There exists a recent study [5] that used convolutional neural network (CNN) with 20 manually crafted features to predict the target of a miRNA. In [5], however, the deep learning method was used as a classifier rather than a feature learner from the raw features (*i.e.*, biological sequences). The novelty of our approach comes from its use of deep RNNs without laborious feature engineering: deepTarget performs RNA sequence modeling using autoencoders and learns the interactions between miRNAs and mRNAs using stacked RNNs. To the best of our knowledge, the proposed methodology is one of the first attempts to predict miRNA targets by sequence modeling using deep RNNs.

2.1 Autoencoder

An autoencoder is an artificial neural network used for learning representations. Autoencoders have almost similar structure to that of multilayer perceptrons (MLPs) except that the output layer has the same number of nodes as the input layer. The objective of an autoencoder is to learn a meaningful encoding for a set of data in an unsupervised manner while a MLP is to be trained to predict the target value in a supervised manner.

Typically, an autoencoder consists of two parts, an encoder and a decoder, which can be defined as transitions ϕ and ψ , respectively, given by

$$\phi: \mathcal{S} \mapsto \mathcal{X} \quad \text{and} \quad \psi: \mathcal{X} \mapsto \mathcal{S}$$
 (1)

subject to

$$\underset{\phi,\psi}{\arg\min} \|S - (\psi \circ \phi)S\|^2 \tag{2}$$

for $S \in \mathcal{S}$, where \mathcal{S} is the input space and \mathcal{X} is the mapped feature space.

Various techniques exist to improve the generalization performance of autoencoders. Examples include the denoising autoencoder (DAE) [47], and the variational autoencoder (VAE) [22]. Recently, the VAE concept has become more widely used for learning generative models of data [14].

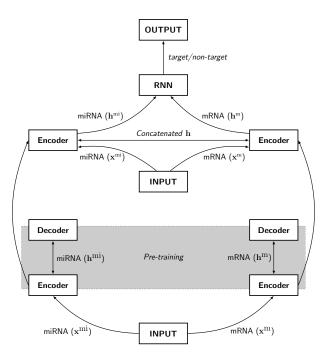


Figure 3: Overview of proposed deepTarget methodology: Our method is based on RNA sequence modeling and sequence-to-sequence interaction learning using autoencoders and stacked RNNs. The input layer is connected to the first layer of two autoencoders to model miRNA and mRNA sequences, respectively. The second layer is an RNN layer to model the interaction between miRNA and mRNA sequences. The outputs of the top RNN layer are fed into a fully connected output layer, which contains two units for classifying targets and non-targets

In this paper, RNN-based autoencoders are used to model miRNA and mRNA sequences to extract their inherent features in an unsupervised manner.

2.2 Recurrent Neural Network (RNN)

An RNN is an artificial neural network where connections between units form a directed cycle. This creates an internal state of the network that allows it to exhibit dynamic temporal behavior. RNNs can use their internal memory to process sequences of inputs. This makes RNNs applicable to various tasks, such as unsegmented connected handwriting recognition [23] and speech recognition [39].

Leveraged by advances in the memory-based units [*e.g.*, long short-term memory (LSTM) [16] and gated recurrent unit (GRU) [6]], network architecture [*e.g.*, stacked RNNs [36] and bidirectional RNNs [40]] and training methods [*e.g.*, iRNN [23] and dropout [43]], RNNs are delivering breakthrough performance in tasks involving sequential modeling and prediction [1, 7, 39, 48].

According to Goodfellow et al. [13], RNNs can do various sequence modeling tasks including the following:

- T1: mapping an input sequence to a fixed-size prediction
- **T2:** modeling a distribution over sequences and generating new ones from the estimated distribution
- T3: mapping a fixed-length vector into a distribution over sequences
- **T4:** mapping a variable-length input sequence to an output sequence of the same length
- **T5:** mapping an input sequence to an output sequence that is not necessarily of the same length

In this paper, RNNs are used to perform task **T5** for sequence modeling using autoencoders and task **T1** for target prediction.

2.3 Biology of miRNA-mRNA Interactions

In molecular biology, directionality of a nucleic acid, 5'-end and 3'-end, is the end-to-end chemical orientation of a single strand of the nucleic acid. The 3' untranslated region (UTR) of mRNA is the region that directly follows the translation termination codon. As shown in Figure 2, miRNAs exhibit their function by binding to the target sites present in the 3' UTR of their cognate mRNAs. By binding to target sites within the 3' UTR, miRNAs can decrease gene expression of mRNAs by either inhibiting translation or causing degradation of the transcript. The seed sequence of a miRNA is defined as the first two to eight nucleotides starting from the 5' to the 3' UTR. The candidate target site (CTS) refers to the small segment of length k of mRNA that contains a complementary match to the seed region at the head. The blue line between a pair of nucleotides (one from miRNA and the other from mRNA) in Figure 2 represents the Watson-Crick (WC) pairing formed by sequence complementarity (i.e., the binding of A-U and that of C-G).

The canonical sites are complementary to the miRNA seed region while the non-canonical sites are with bulges or single-nucleotide loops in the seed region. The non-canonical sites include two types, 3' compensatory and centered sites. The proposed machine learning-based approach has potential to learn both canonical and non-canonical sites. While considering CTSs to predict targets of given miRNAs, there are four types of seed sequence matches [37] often referred to as 6mer, 7mer-m8, 7mer-A1, and 8mer in the literature. The details of each type are as follows:

 6mer: exact WC pairing between the miRNA seed and mRNA for six nucleotides

- 7mer-m8: exact WC pairing from nucleotides 2–8 of the miRNA seed
- 7mer-A1: exact WC pairing from nucleotides 2–7 of the miRNA seed in addition to an A of the miRNA nucleotide 1
- 8mer: exact WC pairing from nucleotides 2–8 of the miRNA seed in addition to an A of the miRNA nucleotide 1

3. PROPOSED METHODOLOGY

Figure 3 shows the overview of the proposed deepTarget methodology. Our method is based on RNA sequence modeling and sequence-to-sequence interaction learning using autoencoders and stacked RNNs. The input layer is connected to the first layer of two autoencoders to model miRNA and mRNA sequences, respectively. The second layer is an RNN layer to model the interaction between miRNA and mRNA sequences. The outputs of the top RNN layer are fed into a fully connected output layer, which contains two units for classifying targets and non-targets.

We preprocess the CTSs with length k, which are empirically determined by all of the four seed matching types between miRNAs and mRNAs as described in Section 2.3. After preprocessing, as shown in Algorithm 1, our approach proceeds in two major steps with miRNA and CTS sequence pair as an input: (1) unsupervised learning of two autoencoders for modeling miRNAs and mRNAs (lines 4–7), and (2) supervised learning of the whole architecture for modeling interaction between miRNAs and mRNAs (lines 10–12).

Throughout this paper, we denote a sequence as \mathbf{s}^{mi} of n_d^{mi} -dimensional vector for miRNA and \mathbf{s}^{m} of n_d^{m} -dimensional vector for mRNA where n_d^{mi} and n_d^{m} can, in general, be different and $k \approx \max(n_d^{\mathrm{mi}})$.

3.1 Input Representation

Each RNA read is a sequence that has four types of nucleotides $\{A, C, G, U\}$ and needs to be converted into numerical representations for machine learning.

A widely used conversion technique is using one-hot encoding [2], which converts the nucleotide in each position of a DNA sequence of length n_d into a four-dimensional binary vector and then concatenates each of the n_d four-dimensional vectors into a $4n_d$ -dimensional vector representing the whole sequence. For instance, let $s \in \mathcal{S}$ where $\mathcal{S} = \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{U}\}$, then, a sequence $\mathtt{s} = (\mathtt{A},\mathtt{G},\mathtt{U},\mathtt{U})$ where $n_d = 4$ is encoded into a tuple of four 4-dim binary vectors:

$$\langle [1,0,0,0], [0,0,1,0], [0,0,0,1], [0,0,0,1] \rangle.$$

According to recent studies [24, 25], however, applying the vanilla one-hot encoding scheme tends to give limited generalization performance caused by the sparsity of the encoding. In lieu of the one-hot encoding scheme, our approach thus encodes each nucleotide into a four-dimensional dense vector that is randomly initialized and trained by the gradient descent method through the whole architecture as a normal neural network layer [8]. We encode sequences to \mathbf{x}^{mi} for miRNA and \mathbf{x}^{m} for mRNA, where \mathbf{x} is a tuple of n_d four-dimensional dense vectors. For instance, a one-hot encoded tuple $\langle [0,0,0,1], [0,0,1,0] \rangle$ is encoded into a dense tuple

$$\langle [-0.1, -0.2, 0.1, 0.3], [-0.1, -0.2, 0.3, 0.4] \rangle.$$

3.2 Modeling RNAs using RNN based Autoencoder

Menor *et al.* [31] described 151 site-level features between miRNA-target pairs for target prediction, categorizing them into seven groups: binding energy, the type of seed matching, miRNA pairing,

target site accessibility, target site composition, target site conservation, and the location of target sites. To relieve these feature engineering required for target prediction in conventional approaches, deepTarget exploits the unsupervised feature learning using the RNN encoder-decoder model [44].

Each of our autoencoder model consists of two RNNs, one as an encoder and the other as a decoder. The encoder RNN encodes \mathbf{x} to \mathbf{h} , and the following decoder RNN decodes \mathbf{h} to reconstructed $\hat{\mathbf{x}}$, which minimizes the reconstruction error

$$\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \tag{3}$$

where \mathbf{h} is a tuple of n_d n_h -dimensional vectors (here n_h is the number of the hidden units of the encoder). Through unsupervised learning of these autoencoders, we get the representations of inherent features that will be used by the stacked RNN layer described in Section 3.3.

After unsupervised learning of these two autoencoders for miRNA and mRNA, respectively, we bypass the decoder and connect the encoder directly to the next layer as a tuple of the fixed dimension n_h representations as shown in Figure 3. This restriction of the fixed dimension n_h is to model the activations of the second layer to be analogous to target pairing patterns. We then obtain sequence representations \mathbf{h}^{mi} for miRNA and \mathbf{h}^{m} for mRNA.

3.3 Modeling Interaction between RNAs

The miRNA-mRNA binding sites can be classified into three types [30]: 5'-dominant canonical, 5'-dominant seed only, and 3'-compensatory. Regardless of the type of a binding site, its detection typically requires the sequence alignment procedure in most of the conventional approaches. However, sequence alignment may often limit the robustness of target prediction because of the need to tune various alignment parameters. The use of an RNN-based approach allows us to omit the sequence alignment procedure (Refer to Figure 7 to see that our RNN-based approach can still detect the binding sites without any alignment).

In this interaction modeling step, the learned features of miRNAs \mathbf{h}^{mi} and mRNAs \mathbf{h}^{m} (extracted by the two autoencoders described in Section 3.2) need to be combined into one tuple \mathbf{h} . To this end, we can devise various ways to merge the representations between each model (*e.g.*, summation, multiplication, concatenation, and average).

The concatenated representations from different modalities have been unsuccessful in constructing a high-dimensional vector since the correlation between features in each modality is stronger than that between modalities [35]. To resolve this issue, deep learning methods have been proposed to learn joint representations that are shared across multiple modalities after learning modality-specific network layers [42]. In this paper, we adopt the method of concatenating each dimension, \mathbf{h}^{mi} and \mathbf{h}^{m} , into tuple \mathbf{h} given by

$$\mathbf{h} = \langle \mathbf{h}_1, \cdots, \mathbf{h}_{n_d} \rangle = \langle (\mathbf{h}_1^{\text{mi}}, \mathbf{h}_1^{\text{m}}), \cdots, (\mathbf{h}_{n_d}^{\text{mi}}, \mathbf{h}_{n_d}^{\text{m}}) \rangle. \tag{4}$$

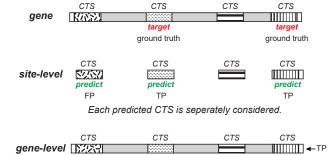
The two unsupervised-learned encoders followed by a stacked RNN layer comprises our proposed architecture for target prediction that has \mathbf{x}^{mi} and \mathbf{x}^{m} as the inputs. The output node has an activation probability given by

$$P(Y = i|\mathbf{h}) = \frac{1/(1 + \exp(-\mathbf{w}_i^T \mathbf{h}))}{\sum_{k=0}^{1} 1/(1 + \exp(-\mathbf{w}_k^T \mathbf{h}))}$$
 (5)

where y is the label whether the given pair is a miRNA-target pair (y=1) or not (y=0), and $k\in\{0,1\}$ is the class index. The

Algorithm 1 Pseudo-code of deepTarget

- 1: **Input:** N encoded RNA sequences, $\mathbf{x}_1^{\text{mi}}, \dots, \mathbf{x}_N^{\text{mi}}$ and $\mathbf{x}_1^{\text{m}}, \dots, \mathbf{x}_N^{\text{m}}$
- 2: **Output:** y (target/non-target)
- 3: Pre-train the autoencoders AE^{mi} and AE^m
- 4: repeat
- 5: minimize the reconstruction error $\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}}) = \|\mathbf{x} \hat{\mathbf{x}}\|^2$
- 6: **until** # of epoch is n_{epoch}
- 7: Define the whole architecture [EN^{mi} || EN^m]-RNN
- 8: Concatenate two representations, \mathbf{h}^{mi} and \mathbf{h}^{m} , into \mathbf{h}
- 9: Fine-tune the architecture
- 10: repeat
- 11: minimize the logarithmic loss $\mathcal{L}(\mathbf{w}) = -\frac{1}{N} \sum_{i=1}^{N} (p_i \log(p_i) + (1-p_i) \log(1-p_i))$
- 12: **until** # of epoch is n_{epoch}



At least one of the predicted CTSs is considered.

Figure 4: Definition of site-level and gene-level target pair datasets: Assume that there are four CTSs on a genome, and only two of them are true target sites (labeled 'target' in the top figure). Suppose that a target prediction tool predicts the first, second, and the fourth CTS as potential targets (labeled 'predict' in the middle and bottom figures). A site-level dataset has a label for each CTS, and we can use it to evaluate the tool performance at site level. By contrast, a gene-level dataset has a level for each gene, and we can evaluate the performance of the prediction tool only at gene level.

objective function $\mathcal{L}(\mathbf{w})$ that has to be minimized is then as follows:

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N} \sum_{i=1}^{N} (p_i \log(p_i) + (1 - p_i) \log(1 - p_i))$$
 (6)

where p_i is the probability of the given pair $(\mathbf{x}^{\text{mi}}, \mathbf{x}^{\text{m}})$ being a valid miRNA-target pair and N is the mini-batch size used.

4. EXPERIMENTAL RESULTS

4.1 Experiment Setup

4.1.1 Dataset

In the literature, there exist two types of datasets that contain miRNA-mRNA pairing information: *site-level* and *gene-level* datasets. As depicted in Figure 4, let us assume that there are four CTSs on a gene sequence, two of which are true target sites of a miRNA. A site-level dataset has a label for each CTS indicating whether this CTS is a target or not, and we can evaluate the performance of a target prediction tool for each CTS. On the other hand, a gene-level dataset has a label for each gene, not for each CTS, and we can

evaluate the prediction performance at the gene level. Refer to the caption for Figure 4 for additional details.

⊳ AE^{mi}, AE^m: the autoencoders described in Section 3.2

▷ EN^{mi}, EN^m: the encoders in described in Section 3.3

 $\triangleright \mathbf{h}^{mi}$, \mathbf{h}^{m} : the outputs of each encoder

To test the proposed deepTarget, we utilized a public human miRNA-mRNA pairing data repository [31], which provides both site-level and gene-level datasets. The creators of this repository obtained validated target sites from miRecords [49] database and mature human miRNA sequences from mirBase [15] database. We used 2,042 human miRNAs that are known to bind to their cognate mRNAs. To train deepTarget, we utilized the 507 site-level and 2,891 gene-level miRNA-target site pairs as the positive training set. The negative training set was generated as explained in the next subsection.

4.1.2 Negative Training Data Generation

The negative dataset was generated using mock miRNAs in a similar manner to [18, 29] that did not have overlap with the seed sequences of any real mature miRNAs in mirBase. The mock miRNAs were generated with random permutaions using the Fisher-Yates shuffle algorithm [10]. The 507 for site-level and 3,133 for gene-level negative mock miRNA-target pairs were then generated for each real miRNA-target pair in the positive dataset by replacing the positive miRNA in the real miRNA-target pair. The negative target regions were generated for each mock miRNA-target pair using MiRanda [9] by thresholding the minimum alignment score for minimizing the biases relevant to the miRanda algorithm.

Note that the negative data generation procedure described above is widely used in the literature to handle lack of biologically meaningful and experimentally verified negative pairs for training. For example, the mock miRNA-based negative dataset we used is essentially identical to that used in [18, 29, 31]. In addition, as mentioned in [31], the mock miRNA-based negative dataset in lieu of real biological negative ones was chosen to obtain a balanced training dataset. These authors investigated four combinations of training and test datasets with the real and mock sequences and found that the mock miRNAs for both model building and validation had the best predictive performance.

4.1.3 Network Architecure and Training

In the following sections, the architecture of our method is denoted by the number of nodes in each layer. For example, an architecture 4-60-30-2 has four layers with four input units, 60 units in the first hidden layer, and so on.

The proposed RNN-based approach used [(4-30-4) || (4-30-4)]-(60-30)-2 where (4-30-4) is for the autoencoder described in Section 3.2 and (60-30) is for the stacked RNN described in Secion 3.3. Note that the symbol '||' represents the two parallel autoencoders. The number of output units of each encoder for miRNA and mRNA,



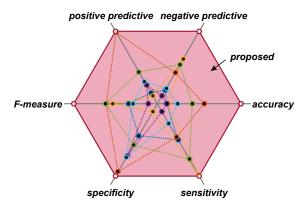


Figure 5: Comparison of prediction performance of proposed deepTarget and alternatives (best viewed in color). The proposed deepTarget outperforms the compared alternatives in terms of all the evaluation metrics used (F-measure, specificity, sensitivity, accuracy, negative predictive value, and positive predictive value; see the footnote on page 5 for the definitions of these metrics and Table 1 for the values of each metric)

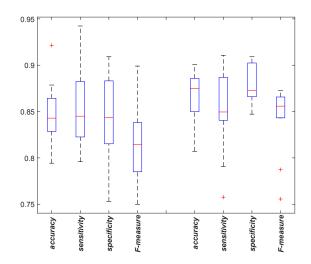


Figure 6: Test accuracy of our approach using two types of memory units (left: deep RNN with LSTM units, right: with GRU units)

Table 1: Comparison of prediction performance of proposed deepTarget and alternatives

	MBSTAR	miRanda	PITA	RNA22	TargetScan	TargetSpy	deepTarget
accuracy	0.5805	0.7737	0.5870	0.5820	0.7125	0.6364	0.9641
sensitivity	0.5308	0.6122	0.9301	0.4006	0.7980	0.4934	0.9389
specificity	0.5308	0.9228	0.2703	0.7494	0.6336	0.7657	0.9706
F-measure	0.5921	0.7220	0.6837	0.4791	0.7271	0.5672	0.9105
PPV	0.5551	0.8797	0.5405	0.5960	0.6677	0.6616	0.8848
NPV	0.6114	0.7206	0.8074	0.5753	0.7727	0.6223	0.9845

respectively, was set to an equal number in order to model the activations of the second layer to be analogous to target pairing patterns.

For training, our approach optimized the logarithmic loss function using Adam [21] [batch size: 50, the number of epochs: 50 (pre-training autoencoder), and 400 (fine-tuning architecture)]. The weights were initialized according to a uniform distribution as directed in [12]. We used dropout as the regularizer in lieu of recently popularized batch normalization [17] with 1,014 site-level target pairs.

The effects of varying architectures and dropout parameters will be presented in Section 4.3.

4.2 Prediction Performance

To evaluate the prediction performance of our deepTarget approach, we compared its performance with six existing tools for miRNA target prediction: MBSTAR [3], miRanda, PITA [20], RNA22 [34], TargetScan [26], and TargetSpy [46] with 6,024 genelevel target pairs. After we found CTSs in 6,024 gene-level target pairs, there remained 4,735 positive and 1,225 negative pairs because of the disappearance of the seed in the mock miRNAs that match the mRNA sequences.

As a performance metric, we used the widely used measures

including accuracy 1 , sensitivity 2 , specificity 3 , F-measure 4 , positive predictive value (PPV) 5 , and negative predictive value (NPV) 6 .

According to the definition of these metrics, they generally have FN in the denominator. Namely, high sensitivity and NPV represent that FN is low; however, the algorithms that classify most of the samples as positive would also have low FN. To distinguish such poor algorithms from more accurate ones, we also have to consider the specificity and PPV metrics, whose definitions include FP in the numerator.

Figure 5 and Table 1 show the comparison of prediction performance. To obtain the performance of deepTarget, we performed 10-fold cross validation and averaged each resulting value. Overall, our method significantly outperformed the best alternative in terms of accuracy, sensitivity, and F-measure by 24.61%, 17.66%, and 25%, respectively. As will be discussed in Section 5, biologists tend to give more priority to sensitivity than to specificity, when

¹accuracy = (TP+TN)/(TP+TN+FP+FN), where TP, FP, FN, and TN represent the numbers of true positives, false positives, false negatives, and true negatives, respectively.

²sensitivity = TP/(TP + FN)

 $^{^{3}}$ specificity = TN/(TN + FP)

 $^{^{4}}$ F-measure = 2TP/(2TP + FP + FN)

 $^{^{5}}PPV = TP/(TP + FP)$

 $^{^{6}}$ NPV = TN/(TN + FN)

Table 2: Effects of architecture variation on prediction performance of deepTarget

Architecture	1-layer	2-layer	3-layer	bidirectional
accuracy	0.8214	0.8286 0.8351	0.8250	0.8250
sensitivity	0.8429		0.8417	0.8254
specificity	0.8204	0.8428	0.8297	0.8041
F-measure	0.7745	0.7838	0.7759	0.7620

Table 3: Effects of changing dropout probability on prediction performance of deepTarget

Dropout	0.0	0.1	0.2	0.3	0.4
accuracy	0.8214	0.8429	0.8179	0.8179	0.7857
sensitivity	0.8429	0.8480	0.8460	0.8460	0.7980
specificity	0.8204	0.8486	0.8111	0.8111	0.7746
F-measure	0.7745	0.7978	0.7592	0.7592	0.7386

searching for potential targets of specific miRNA [28].

4.3 Effects of Architecture Variation

Figure 6 shows the effects of the memory unit used in deepTarget on its prediction performance. To generate this plot, we performed 10-fold cross validation and tested two popular memory units, LSTM and GRU. According to Jozefowicz *et al.* [19], it is believed that there is no architecture that can consistently beat the LSTM and the GRU in all conditions.

As shown in Figure 6, the architecture composed of GRU outperforms that of LSTM in our experiments. More quantitatively, the architecture using GRU as its memory unit showed 3.81%, 0.53%, 3.46%, and 5.06% improvements over the LSTM-installed architecture in terms of accuracy, sensitivity, specificity, and F-measure, respectively (the median values achieved were 0.8750, 0.8496, 0.8728, and 0.8559, respectively, in 10-fold cross validation).

Table 2 lists the effect of varying the architecture of the second layer RNN on overall performance. We tested stacked RNNs with one to three layers and single-layer bidirectional RNNs. Using the two layer architecture gave the best overall results, although the differences in performance were not significant.

Table 3 presents the effect of changing the probability of dropout on performance. The architecture with dropping hidden unit activations with a probability of 0.1 during training gave the best performance, delivering 0.61–3.44% improvements over the other probability values tried.

4.4 Visual Inspection of RNN Activations

Figure 7 shows the activations in the RNN layer in the second stage for learning sequence-to-sequence interactions. The x-axis represents two RNA sequences (top: miRNA, bottom: mRNA), and the y-axis represents the position of each hidden unit in the layer. Using this plot, we can see how the activation changes over the nucleotide positions along the RNA sequences. Note that darker shades represents higher activation values. The red box indicates the activation sequence of the hidden unit that best represents the binding site pattern. This pattern seems compatible with the sequence alignment results shown by the two aligned RNA sequences in the x-axis. These results suggest that analyzing the activation patterns may allow us to gain novel biological insight into regulatory interactions.

5. DISCUSSION

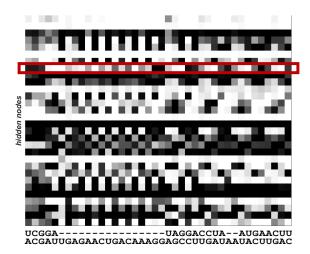


Figure 7: Change of activation of each node in the RNN layer over RNA sequences [darker shades mean higher activation values; the red box indicates the activation sequence of the hidden unit that best represents the binding site pattern]

Biological sequences (such as DNA, RNA, and protein sequences) naturally fit the recurrent neural networks that are capable of temporal modeling. Nonetheless, prior work on applying deep learning to bioinformatics utilized only convolutional and fully connected neural networks. The biggest novelty of our work lies in its use of recurrent neural networks to model RNA sequences and further learn their sequence-to-sequence interactions, without laborious feature engineering (e.g., more than 151 features of miRNA-target pairs have been proposed in the literature). As shown in our experimental results, even without any of the known features, deepTarget delivered substantial performance boosts (over 25% increase in F-measure) over existing miRNA target detectors, demonstrating the effectiveness of recent advances in end-to-end learning methodologies.

When dealing with genome-wide applications, such as miRNA target prediction, we often need to handle extremely imbalanced datasets in which negative examples significantly outnumber positive examples. According to Saito and Rehmsmeier [38], as PPV changes with the ratio of positives and negatives, PPV is more useful than the other metrics for evaluating binary classifiers on imbalanced datasets. As mentioned in Section 4.2, deepTarget was trained with 4,735 positive and 1,225 negative pairs and showed higher performance than the alternatives in terms of PPV. The higher values of sensitivity and PPV deepTarget reported from imbalance datasets indicate that deepTarget performs more robust prediction of miRNA-mRNA pairs.

When training deepTarget, we focused on improving its capability to reject false positives (*i.e.*, bogus miRNA-mRNA pairs) as a target predictor. Given that there is a trade-off between sensitivity and specificity, our training principle resulted in slight loss in the specificity of deepTarget. This decision was based on the study that more priority should be given to sensitivity in the search for potential targets of specific miRNAs, whereas specificity should be emphasized in the examination of miRNAs that regulate specific genes [28]. Depending on the specific needs, we could alternatively train deepTarget to put more priority on specificity. For instance, this could be done by altering the composition of a mock negative dataset to have additional mispairings between miRNA and mRNA sequences except the seed sequence.

Notably, deepTarget does not depend on any sequence alignment operation, which has been used in many bioinformatics pipelines as a holy grail to reveal similarity/interactions between sequences. Although effective in general, sequence alignment is susceptible to changes in parameters (e.g., gap/mismatch penalty and match premium) and often fails to reveal the true interactions between sequences, as is often observed in most of the alignment-based miRNA target detectors. By processing miRNA and RNA sequences with RNN-based auto-encoders without alignment, deepTarget successfully discover the inherent sequence representations, which are effectively used in the next step of deepTarget for interaction learning.

In Figure 7, we visualized the activations in the RNN layers of deepTarget as an attempt to discriminate between the positive and negative samples. Contrasting and scrutinizing the patterns existing in each class will be helpful for providing insight that can eventually lead to discovery of novel biological features. Given that biomedical practitioners prefer 'white-box' approaches (e.g., logistic regression) to 'black-box' approaches (e.g., deep learning), such efforts will hopefully expedite the deployment and acceptance of deep learning for biomedicine [33]. In this regard, attempts to decode the information learned by deep learning are emerging (e.g., we can compute importance scores for each activation to better interpret activation energies [41]).

Although the performance of deepTarget is incomparably higher than that of the existing tools compared, there still remains room for further improvements. An additional breakthrough may be possible by enhancing the current step to learn sequence-to-sequence interactions. The current version of deepTarget relies on concatenating the RNA representations from two auto-encoders and learning interactions therein using a unidirectional two-layer RNN architecture. Although this architecture was effective to some extents, as shown in our experiments, adopting even more sophisticated approaches may further boost the capability of deepTarget to detect subtle interactions that currently go undetected. Our future work thus includes investigating how to improve the interaction learning part, which is crucial for achieving additional performance gains.

Acknowledgment

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science, ICT and Future Planning) [No. 2012M3A9D1054622, No. 2014M3C9A3063541, No. 2014M3A9E2064434, and No. 2015M3A9A7029735], and in part by the Brain Korea 21 Plus Project in 2016.

6. REFERENCES

- [1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv* preprint arXiv:1409.0473, 2014.
- [2] P. Baldi and S. Brunak. Chapter 6. neural networks: applications. In *Bioinformatics: The Machine Learning Approach*. MIT press, 2001.
- [3] S. Bandyopadhyay, D. Ghosh, R. Mitra, and Z. Zhao. MBSTAR: multiple instance learning for predicting specific functional binding sites in microRNA targets. *Scientific* reports, 5, 2015.
- [4] D. P. Bartel. MicroRNAs: genomics, biogenesis, mechanism, and function. *Cell*, 116(2):281–297, 2004.
- [5] S. Cheng, M. Guo, C. Wang, X. Liu, Y. Liu, and X. Wu. MiRTDL: a deep learning approach for miRNA target prediction.
- [6] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder

- approaches. arXiv preprint arXiv:1409.1259, 2014.
- [7] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078, 2014.
- [8] F. Chollet. Keras: Deep Learning library for Theano and TensorFlow. https://github.com/fchollet/keras, 2015.
- [9] A. J. Enright, B. John, U. Gaul, T. Tuschl, C. Sander, D. S. Marks, et al. MicroRNA targets in Drosophila. *Genome Biology*, 5(1):R1–R1, 2004.
- [10] R. A. Fisher, F. Yates, et al. Statistical tables for biological, agricultural and medical research. Statistical tables for biological, agricultural and medical research., (Ed. 3.), 1949.
- [11] R. C. Friedman, K. K.-H. Farh, C. B. Burge, and D. P. Bartel. Most mammalian mRNAs are conserved targets of microRNAs. *Genome Research*, 19(1):92–105, 2009.
- [12] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International* conference on artificial intelligence and statistics, pages 249–256, 2010.
- [13] I. Goodfellow, Y. Bengio, and A. Courville. Deep learning. Book in preparation for MIT Press, 2016.
- [14] K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. DRAW: A recurrent neural network for image generation. arXiv preprint arXiv:1502.04623, 2015.
- [15] S. Griffiths-Jones, H. K. Saini, S. van Dongen, and A. J. Enright. miRBase: tools for microRNA genomics. *Nucleic Acids Research*, 36(suppl 1):D154–D158, 2008.
- [16] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [17] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [18] B. John, A. J. Enright, A. Aravin, T. Tuschl, C. Sander, D. S. Marks, et al. Human microRNA targets. *PLoS Biol*, 2(11):e363, 2004.
- [19] R. Jozefowicz, W. Zaremba, and I. Sutskever. An empirical exploration of recurrent network architectures. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 2342–2350, 2015.
- [20] M. Kertesz, N. Iovino, U. Unnerstall, U. Gaul, and E. Segal. The role of site accessibility in microRNA target recognition. *Nature Genetics*, 39(10):1278–1284, 2007.
- [21] D. Kingma and J. Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.
- [22] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [23] Q. V. Le, N. Jaitly, and G. E. Hinton. A simple way to initialize recurrent networks of rectified linear units. arXiv preprint arXiv:1504.00941, 2015.
- [24] B. Lee, T. Lee, B. Na, and S. Yoon. DNA-level splice junction prediction using deep recurrent neural networks. *arXiv* preprint arXiv:1512.05135, 2015.
- [25] T. Lee and S. Yoon. Boosted categorical restricted boltzmann machine for computational prediction of splice junctions. In *ICML*, pages 2483âĂŞ–2492, 2015.
- [26] B. P. Lewis, I.-h. Shih, M. W. Jones-Rhoades, D. P. Bartel, and C. B. Burge. Prediction of mammalian microRNA targets. *Cell*, 115(7):787–798, 2003.
- [27] J. Lu, G. Getz, E. A. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B. L. Ebert, R. H. Mak, A. A.

- Ferrando, et al. MicroRNA expression profiles classify human cancers. *nature*, 435(7043):834–838, 2005.
- [28] T. M Witkos, E. Koscianska, and W. J Krzyzosiak. Practical aspects of microRNA target prediction. *Current molecular medicine*, 11(2):93–109, 2011.
- [29] M. Maragkakis, P. Alexiou, G. L. Papadopoulos, M. Reczko, T. Dalamagas, G. Giannopoulos, G. Goumas, E. Koukis, K. Kourtis, V. A. Simossis, et al. Accurate microRNA target prediction correlates with protein repression levels. *BMC Bioinformatics*, 10(1):295, 2009.
- [30] P. Maziere and A. J. Enright. Prediction of microRNA targets. Drug discovery today, 12(11):452–458, 2007.
- [31] M. Menor, T. Ching, X. Zhu, D. Garmire, and L. X. Garmire. mirMark: a site-level and UTR-level classifier for miRNA target prediction. *Genome biology*, 15(10):500, 2014.
- [32] H. Min and S. Yoon. Got target?: Computational methods for microRNA target prediction and their extension. *Experimental* & *Molecular Medicine*, 42(4):233–244, 2010.
- [33] S. Min, B. Lee, and S. Yoon. Deep learning in bioinformatics. *Briefings in Bioinformatics*, in press, 2016.
- [34] K. C. Miranda, T. Huynh, Y. Tay, Y.-S. Ang, W.-L. Tam, A. M. Thomson, B. Lim, and I. Rigoutsos. A pattern-based method for the identification of microRNA binding sites and their corresponding heteroduplexes. *Cell*, 126(6):1203–1217, 2006.
- [35] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 689–696, 2011.
- [36] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio. How to construct deep recurrent neural networks. arXiv preprint arXiv:1312.6026, 2013.
- [37] S. M. Peterson, J. A. Thompson, M. L. Ufkin, P. Sathyanarayana, L. Liaw, and C. B. Congdon. Common features of microRNA target prediction tools. *Front Genet*, 5:23, 2014.
- [38] T. Saito and M. Rehmsmeier. The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. *PloS One*, 10(3):e0118432, 2015.
- [39] H. Sak, A. Senior, and F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Proceedings of the Annual Conference of*

- International Speech Communication Association (INTERSPEECH), 2014.
- [40] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, 45(11):2673–2681, 1997.
- [41] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje. Not just a black box: Learning important features through propagating activation differences. arXiv preprint arXiv:1605.01713, 2016.
- [42] K. Sohn, W. Shang, and H. Lee. Improved multimodal deep learning with variation of information. In *Advances in Neural Information Processing Systems*, pages 2141–2149, 2014.
- [43] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
- [44] N. Srivastava, E. Mansimov, and R. Salakhutdinov. Unsupervised learning of video representations using LSTMs. arXiv preprint arXiv:1502.04681, 2015.
- [45] P. K. Śrivastava, T. R. Moturu, P. Pandey, I. T. Baldwin, and S. P. Pandey. A comparison of performance of plant miRNA target prediction tools and the characterization of features for genome-wide target prediction. *BMC Genomics*, 15(1):1, 2014.
- [46] M. Sturm, M. Hackenberg, D. Langenberger, and D. Frishman. TargetSpy: a supervised machine learning approach for microRNA target prediction. *BMC Bioinformatics*, 11(1):292, 2010
- [47] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international* conference on Machine learning, pages 1096–1103. ACM, 2008
- [48] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. arXiv preprint arXiv:1411.4555, 2014.
- [49] F. Xiao, Z. Zuo, G. Cai, S. Kang, X. Gao, and T. Li. miRecords: an integrated resource for microRNA–target interactions. *Nucleic Acids Research*, 37(suppl 1):D105–D110, 2009.
- [50] S. Yoon and G. De Micheli. Computational identification of microRNAs and their targets. *Birth Defects Research Part C: Embryo Today: Reviews*, 78(2):118–128, 2006.