공학석사 학위논문

자기참조 모델링을 통한 전사인자의 서열 특이성 예측

Predicting Sequence Specificities of Transcription Factors with Self-Attention Sequence Modeling

2019 년 2 월

서울대학교 대학원

공과대학 컴퓨터공학부

안 용 주

자기참조 모델링을 통한 전사인자의 서열 특이성 예측

Predicting Sequence Specificities of Transcription Factors with Self-Attention Sequence Modeling

지도교수 김 선

이 논문을 공학석사 학위논문으로 제출함

2018 년 10 월

서울대학교 대학원

공과대학 컴퓨터공학부

안 용 주

안용주의 공학석사 학위논문을 인준함

2018 년 12 월

위 원 장	Srinivasa Rao Satti	(인)
부위원장	김 선	(인)
위 원	Bernhard Egger	(인)

Abstract

Predicting Sequence Specificities of Transcription Factors with Self-Attention Sequence Modeling

Yongjoo Ahn
Department of Computer Science & Engineering
College of Engineering
The Graduate School
Seoul National University

Transcription factor plays crucial role in gene expression via regulating transcription process. To predict the sequence specificity of each transcription factor I propose a deep learning model, the AttendBind which employs k-mer embedding and self-attention sequence modeling approaches. The experimental results on real biophysical data show that the proposed method outperforms other deep learning methods, indicating that the self-attention

sequence modeling is highly effective on this task. In addition to the given prediction task, the visualization of self-attention maps and top-3 frequency based analyses can provide useful information for interpreting the deep learning model and discovering scientific knowledge.

Keywords: Transcription Factor, Self-Attention, Protein Binding Array, Sequence Modeling, Deep Learning

Student Number: 2017-24922

Contents

List of Figures

List of Tables

Chapter 1

Introduction

To express the information in genes of cells, the role of various DNA-binding proteins is crucial. This group of protein takes charge of important cell functions such as DNA and RNA synthesis, DNA repair and cleaving, chromosome packaging, and modulating gene expressions. As its name implies, a DNA-binding protein has binding sites along the DNA and it has a specific or general affinity for the sites. Proteins like polymerases which synthesize DNA or RNA, or histones which are involved in DNA condensation are classified as non-specific binding proteins because they behave DNA sequence-agnostic manner. On the other hand, proteins called transcription factors have sequence specificities which make them bind to specific DNA sequences [1]. A transcription factor (TF) is a protein that modulates gene expressions via controlling the rate of transcription of genetic information from DNA to RNA. It makes genes to be expressed at the right time and for the right amount thus orchestrate many essential cell activities likes hormone response, cell division, cell growth or cell death.

To characterize the sequence specificity of a transcription factor, the binding affinity between the protein and DNA sequences should be determined. Because the TF-DNA binding affinity is a key to quantitative understanding

of sequence specificity, various biophysical experiments have been designed to measure it. Especially the recent developments of high-throughput biotechnology have guided significant experimental methods like protein binding microarray (PBM) or chromatin immunoprecipitation sequencing (ChIP-seq) to be proposed to tackle this problem [2, 3]. The data from these high-throughput methods is too massive and noisy to be interpreted by biologists, thus require data-driven computational approaches such as word count based methods or probabilistic methods like hidden markov model (HMM) [4, 5].

Deep learning has achieved the state-of-the-art performance in diverse machine learning domains, especially in image and natural language processing. The DeepBind [6] demonstrated the significant success of deep learning method when it comes to bioinformatics domain. It used convolutional neural network (CNN) to predict the sequence specificities of DNA-binding proteins. Since the success of CNN based method, the DeeperBind [7] proposed a hybrid neural network which stacked recurrent neural network (RNN) upon CNN architecture and showed the capability of RNN.

RNN coupled with attention mechanism have been successful in sequence modeling domain [8]. Recently, the Transformer [9], which leverages self-attention mechanism, was proposed to address the inherent sequential constraint of RNN and it has been achieving state-of-the-art performance in various sequence modeling tasks.

In this work, I propose the AttendBind, a deep learning model introducing self-attention with k-mer embedding to task of predicting the binding affinity of a transcription factor and DNA sequences. Firstly, the AttendBind interpret given DNA sequences as consecutive k-mers and transforms into sequence of real-value vectors via an embedding layer. Then it equips attention information within the sequence by self-attention encoder. Finally, the vector sequence is concatenated into a single vector and used in downstream neural network to predict the corresponding TF-DNA binding affinity. Experimental results on real data show that the performance of the proposed approach in terms of correlation coefficient and area under the curve (AUC) metrics is better than baselines. Besides, the visualization of computed attention maps from self-attention encoder gives great interpretation for the deep learning model, which well agrees with known biological facts.

The remaining part of this paper is organized as follows. Chapter 2 describes the PBM data used in this work, attention mechanism and the Transformer model, and the details of the proposed AttendBind. Chapter 3 shows experiment results and the last chapter discuss the capability of proposed model.

Chapter 2

Methods

This chapter starts with the description of used data in this work. The second section gives the clarification of the attention mechanism in neural networks and the following section describes the Transformer network. Finally, the proposed AttendBind is illustrated in the last section.

2.1 Data

Protein binding microarray (PBM) is a DNA microarray-based technology that leverages high-throughput characterization of the *in vitro* DNA binding site specificities of transcription factors [10]. The PBM technology has enabled the profiling of the sequence specificity of a given TF by measuring its binding affinity for DNA probes. The *in vitro* results is in agreement with in vivo genome-wide location analysis (ChIP-chip) [11], thus its confidence has been well recognized.

PBM arrays are constructed by taking a normal microarray and synthesizing a complementary strand for each probe using DNA polymerase. Then

antibody-labeled transcription factor is allowed to bind to probes on the microarray. Each typical PBM data consists of tens of thousands DNA probe sequences and corresponding fluorescence signal intensity scores representing the relative binding affinity of the given TF to probes. Figure 1 shows an example of a PBM data.

Given PBM data, regression problem can be established straightforwardly and a binary classification problem can be set where the positive data is labeled when the signal intensity is larger than the mean plus 4 times standard deviation.

Sequence	Signal_Mean
AAAAAACAACAGGAGGGCATCATGGAGCTGTCCAGCCTGTGTGAAATTGTTATCCGCTCT	290.5074
AAAAAACAGCCGGATCACAATTTTGCCGAGAGCGACCTGTGTGAAATTGTTATCCGCTCT	679.3055
AAAAAACGTCCGGTACACCCCGTTCGGCGGCCCAGCCTGTGTGAAATTGTTATCCGCTCT	1998.715
AAAAAACTCTAGACCTTTAGCCCATCGTTGGCCAACCTGTGTGAAATTGTTATCCGCTCT	447.8039
AAAAAAGAACAACCGGATAACACCCTTACAGCACACCTGTGTGAAATTGTTATCCGCTCT	2846.6899
AAAAAAGCTAAATCTCACTACTATCAACCACGTGCCCTGTGTGAAATTGTTATCCGCTCT	355.98
AAAAAATCGGCGCTCGCACAATAACACTTGGACCACCTGTGTGAAATTGTTATCCGCTCT	862.8706
AAAAAATGGGCGTAAGCGTATTAGGTGGGAACCACCCTGTGTGAAATTGTTATCCGCTCT	307.876
AAAAACATATTTTTAAGCCCCATTGCGATCCAGCTCCTGTGTGAAATTGTTATCCGCTCT	587.8928
AAAAACCCCAGAAAGGTTGACTAGAGTAAATCTCCCCTGTGTGAAATTGTTATCCGCTCT	325.3881
AAAAACCTAAGAATCGTTTCTGAGTCATGAGGTTTCCTGTGTGAAATTGTTATCCGCTCT	456.2666
AAAAACGCGCTGTTTTTCATGCTACACGTCTCAGGCCTGTGTGAAATTGTTATCCGCTCT	1357.5722
AAAAACGTAGTGTGCCGCTGGTAAAGGCTCCGTCCCCTGTGTGAAATTGTTATCCGCTCT	364.5655
AAAAACTAGCTATCTTCGCGTCCACATCCGCCTCACCTGTGTGAAATTGTTATCCGCTCT	443.4153
AAAAACTTTTTACAAGAACTTATGACTTCGACTCGCCTGTGTGAAATTGTTATCCGCTCT	655.0422
AAAAAGAACCCCCGAACAATTAAATCACAAGGGGGCCTGTGTGAAATTGTTATCCGCTCT	370.3642
AAAAAGAACTACCGATGAATGCGCTCTGTTAGTCGCCTGTGTGAAATTGTTATCCGCTCT	639.8161
AAAAAGCCACATCGGGCTTTAAGCCTGGAGCTATTCCTGTGTGAAATTGTTATCCGCTCT	226.8623
AAAAAGCGACATTCTGCCCTTAGTGACTCATGAGGCCTGTGTGAAATTGTTATCCGCTCT	616.2941

Figure 1: **An example of PBM data.**

2.2 Attention Mechanism

Sequence Encoder

Most successful deep learning models dealing with sequence have an encoder that convert input sequence into meaningful representation [12]. With this encoded representation of given sequence, downstream neural network performs task specific computation. If the given task is a sequence transduction classification problem, e.g. machine translation, the decoder computes decoded sequence, and if the task is sequence classification problem, the following network should compute the probability for each class label. In regression task which includes the sequence specificity prediction problem, the downstream network should compute a single real value that should be same as the given ground truth.

Attention Mechanism

Attention mechanism was introduced successfully in sequence transduction task. Sequence-to-sequence (seq2seq) architecture which consists of a recurrent neural network (RNN) encoder and an RNN decoder has been the standard approach to solve neural machine translation problem [12]. Recently the attention mechanism has been integrated with the seq2seq model and considered as an essential component [6].

Given a set of vector values, and a vector query, attention mechanism is a technique to compute a weighted sum of the values dependent of on the query. In case of a seq2seq model, the RNN encoder takes an input sequence of symbol representation $\mathrm{X} = (x_1, x_2, \ldots, x_N)$ and computes encoder hidden states, or values, $h_1, h_2, \ldots, h_N$ where $h_i \in \mathbb{R}^{d_1}$. The decoder hidden state, or a query, $s \in \mathbb{R}^{d_2}$ and the encoder hidden states together make the attention scores $e \in \mathbb{R}^N$. There are several ways to compute $e = (e_1, e_2, \ldots, e_N)$ where $e_i \in \mathbb{R}$:

- Basic dot-product attention:

  $$e_i = s^T h_i \ ,$$

  where it assumes $d_1 = d_2$.

- Multiplicative attention:

  $$e_i = s^T W h_i$$

  where $W \in \mathbb{R}^{d_2 \times d_1}$ is a weight parameter.

- Additive attention:

  $$e_i = v^T \tanh(W_1 h_i + W_2 s)$$

  where $v \in \mathbb{R}^{d_3}$ is a weight vector, $W_1 \in \mathbb{R}^{d_3 \times d_1}$ and $W_2 \in \mathbb{R}^{d_3 \times d_2}$ are weight matrices.

Then the attention distribution $\alpha \in \mathbb{R}^{\mathrm{N}}$ can be computed by softmax function,

$$\alpha = \mathrm{softmax}(e).$$

Given the attention distribution, a weighted sum of the encoder hidden states, $a \in \mathbb{R}^{d_1}$ is computed as follows:

$$a = \sum_{i=1}^{N} \alpha_i h_i.$$

Finally, the decoder hidden states along with the attention output a is fed to downstream neural network which generates an symbol output $\hat{y}$.

The attention mechanism significantly improves the neural machine translation performance because it allows queries, the decoder states, to focus directly on source, the encoder states, and mitigates the bottleneck problem of the final encoder state in source sequence representation. And it gives some interpretability for seq2seq model by inspecting the attention distribution.

2.3 Transformer

Notwithstanding the great success of RNN based sequence modeling such as long short-term memory (LSTM) [13] or gated recurrent unit (GRU) network [14], its inherent sequential property prevents computational parallelization. Also, the path length which is required for learning long-range dependencies among a sequence grows with the length of given sequence.

To solve these problems of RNN, the Transformer [9] which is a seq2seq model that relies solely on attention mechanism was proposed and achieved state-of-the-art performance on many natural language processing (NLP) tasks. Based on the knowledge that attention gives model access to any hidden

state, the Transformer use self-attention to model the given input sequence of symbol representation by relating different positions of a single sequence [9].

Scaled Dot-product Attention

The Transformer use scaled dot-product attention to compute the attention scores. Given a query and a set of key-value pairs, it maps them to an output, where the query, keys, values and output are all vectors. The output is weighted sum of the values, where the weight of each value is computed by the dot product of query and corresponding key.

Given a query $q \in \mathbb{R}^d$, a key matrix K packing keys $k_i \in \mathbb{R}^d$ and a value matrix V packing values $v_j \in \mathbb{R}^d$ where $i, j \in \{1, 2, \dots, N\}$ and N is the size of key-value pairs, the output is defined with attention mapping as follows:

$$\text{Attention}(q, K, V) = \sum_i \frac{e^{q \cdot k_i}}{\sum_j e^{q \cdot k_j}} v_i.$$

When we have multiple queries, it can be stack into a matrix Q, then the outputs are computed as:

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T)V.$$

Figure 2 illustrates this dot-product attention calculation in the Transformer. Each query vector q_i packed in matrix Q makes attention weights with key vectors k_i via matrix multiplication with K^T. The computed weights are normalized with softmax function and makes final output vectors by matrix multiplication with value vectors packed in matrix V.

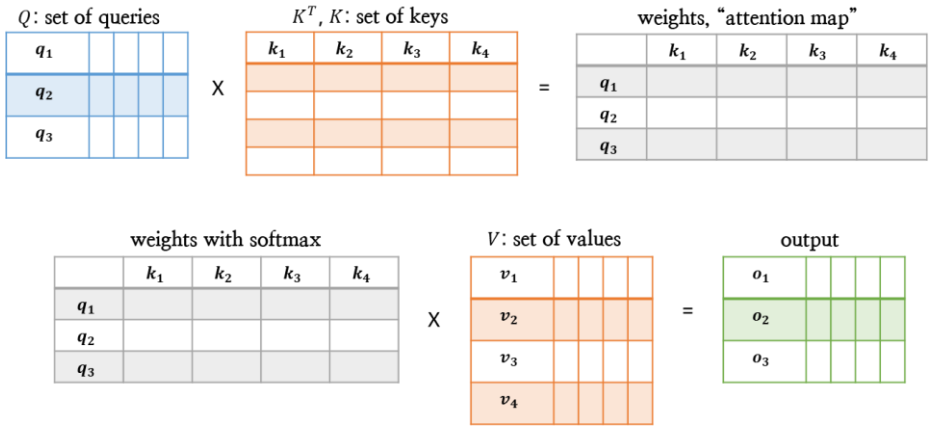

Figure 2: **An illustration of dot-product attention.** The dot-product of vectors can be reformulated by matrix multiplication of query matrix Q and key matrix K^T. Then the computed weights and value vectors V makes the result vectors. In this example, the query, key and value vectors are all 4-dimensional.

For large values of d, the dimensionality of query and key, the variance of $q \cdot k$ increases and the softmax gets peaked causing extremely small gradients. To solve the problem, the final scaled dot-product attention is scaling the softmax by d:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V.$$

Multi-head Attention

In a machine translation task, the input words vectors could be the queries, keys and values of self-attention computation. In other words, the word vectors themselves selectively attention on each other. For a single self-attention, there is only one way for words to interact with one-another and it lacks sequence modeling expressivity. To solve this problem, the multi-head attention was proposed which maps Q, K and V into several lower dimensional spaces then apply attention and concatenates outputs.

2.4 AttendBind

The AttendBind follows the approach of the Transformer model. However, the aforementioned Transformer model is designed for sequence transduction problem, so I adopted the self-attention encoder for generating the representation of sequences and constructed following feed forward neural network for given prediction task. Briefly, the encoder maps a length-N input sequence of symbol representations $\mathrm{S} = (s_1, s_2, \dots, s_N)$ to a sequence of continuous representation $\mathrm{Z} = (z_1, z_2, \dots, z_N)$. Then the downstream neural network views this vector sequence as a single concatenated vector z and generates a real value $\hat{y}$ which is the predicted value of binding affinity between given sequence S and a transcription factor. Figure 2 depicts the overall model architecture of the AttendBind.

Sequence Representation

Existing deep learning methods for predicting sequence specificities of transcription factors use one-hot encoding for representing input DNA sequences. DeepBind transforms DNA sequences into image-like 2-D representation via one-hot encoding and leverages the CNN's capability for processing those form of data. DeeperBind adopts the same sequence representation manner as DeepBind, and augmentatively stacks RNNs over the CNN structure.

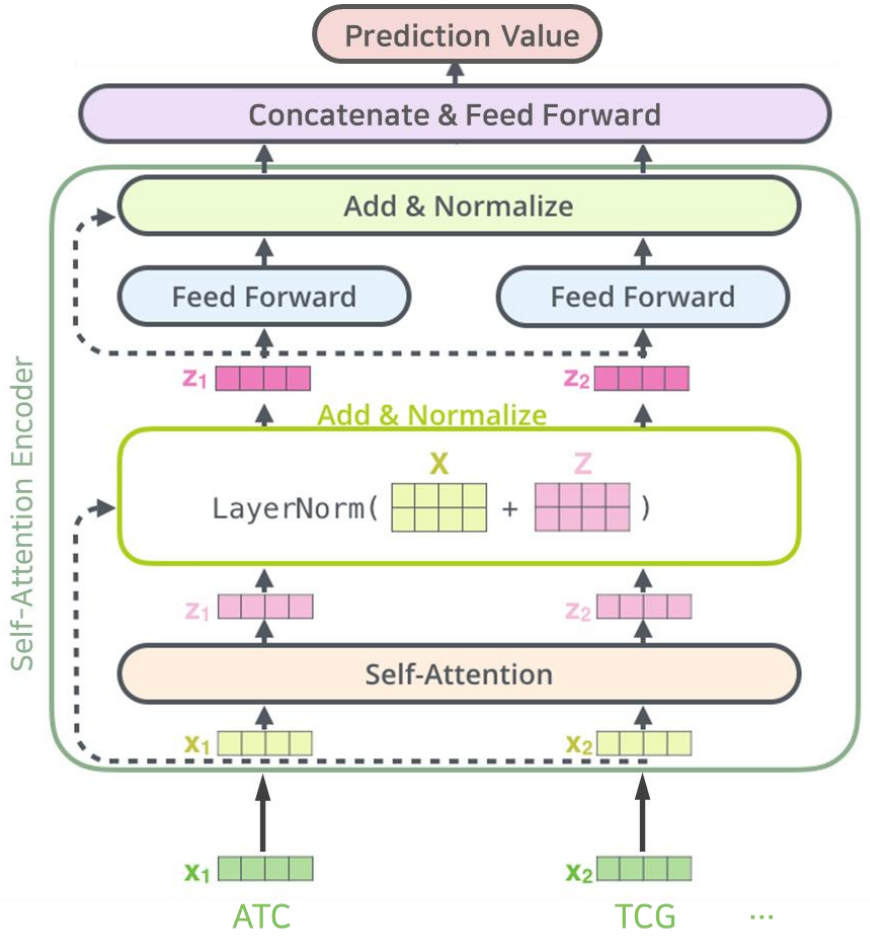

Figure 3: **The overall model architecture of the AttendBind.** Input sequence of k-mers (ATC, TCG, ...) is embedded into $(x_1, x_2, \ldots)$ then encoded by self-attention encoder. The final feed forward network concatenates the encoded vectors and predicts a real value.

In the AttendBind, I use k-mer embedding based representation for DNA sequences. Given a DNA sequence S of length L:

$$\mathrm{S} = (s_1, s_2, \dots, s_L),$$

where $s_i \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}$, and $\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{and}\ \mathrm{T}$ are 4 bases of DNA, it can be interpreted as a sequence of k-mers KS of length $L_k = L - k + 1$ with 1 width stride:

$$\mathrm{KS} = \left(k_1, k_2, \dots, k_{L_k}\right),$$

where $k_i \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^k$.

Then the embedding layer maps k_i into d_{model}-dimensional vector for $i \in \{1, 2, \dots, L_k\}$. Finally, the k-mer embedded sequence representation $\mathrm{X} = (x_1, x_2, \dots, x_N)$ is obtained and fed to the sequence encoder, where $x_i \in \mathbb{R}^{d_{model}}$ is embedded vector of k_i and $N = L_k$.

Self-Attention Encoder

The input embedded vectors could be the queries, keys and values in self-attention making themselves be attended to each other. Given k-mer embedded vectors X, it can be stacked into a matrix form $X \in \mathbb{R}^{d_N \times d_{model}}$, and $Q = K = V = X$ where Q, K, and V are queries, keys and values in attention mapping.

The self-attention encoder performs the multi-headed scaled dot-product

attention as proposed in the Transformer. The multi-head attention splits the d_{model}-dimensional embedded vector into h d_h-dimensional vectors by multiplying weight matrices:

$$Q_i = QW_i^Q, K_i = KW_i^K, V_i = VW_i^V \in \mathbb{R}^{d_N \times d_h} \text{ for } i = 1, \dots, h,$$

where $d_h = \frac{d_{model}}{h}$, Q, K, and V are queries, keys, and values respectively, and $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_{model} \times d_h}$ are parameter weight matrices for splitting dimensionality. Then $head_i$ computed by applying scaled dot-product:

$$head_i = \text{Attention}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i {K_i}^T}{\sqrt{d_h}}\right) V_i.$$

The output Z of self-attention layer is,

$$\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \dots, head_h) W^O,$$

where Concat is vector concatenating function and $W^O \in \mathbb{R}^{d_{model} \times d_{model}}$ is a parameter weight matrix for recovering the full dimensionality.

Figure 4 illustrates the calculation process of multi-head attention in the self-attention encoder. Input vectors X generates queries Q_i, keys K_i and values V_i for each $head_i$ by matrix multiplication with W_i^Q, W_i^K, W_i^V. All heads computed by scaled dot-product attention are concatenated and make final output Z by a linear layer whose weight parameter is W^O.

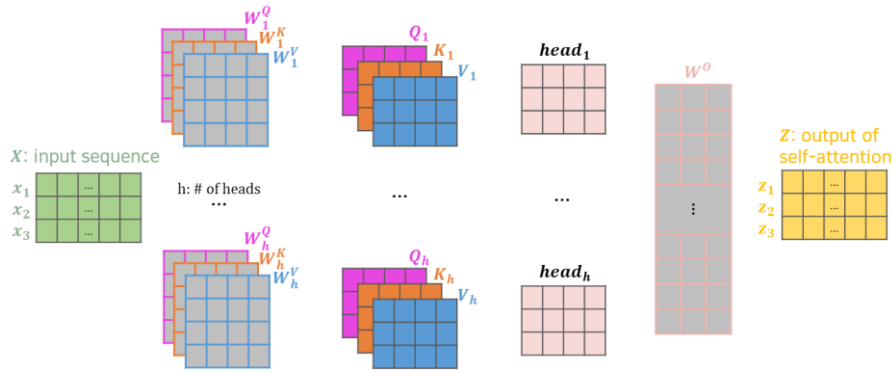

Figure 4: **An illustration of multi-head attention in self-attention encoder.** Q_i, K_i, V_i which are inputs for scaled dot-product attention are made by matrix multiplication of vector sequence X and parameter matrices W_i^Q, W_i^K, W_i^V. Computed heads and W^O then generate the output vectors.

With $Z = (z_1, z_2, \ldots, z_N)$ which is the output of self-attention layer, the encoder applies residual connection [15] and layer normalization [16] as depicted in Figure 3, then feed it to a 2-layer fully connected feed-forward network with ReLU activation function, which is applied each position separately and identically.

TF-DNA Binding Affinity Prediction

$\hat{\mathbf{Z}} = (\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_N)$, which is the final output of the encoder, represents given DNA sequence $\mathbf{S}$. For downstream network, all vectors of $\hat{\mathbf{Z}}$ are concatenated into a single vector $\hat{\mathbf{Z}}_\mathbf{s} = [\hat{z}_1; \hat{z}_2; \ldots; \hat{z}_N]$. To generate a single real

value $\hat{y}$, which is the predicted binding affinity, I use 2-layer fully connected network which takes $\hat{\mathbf{Z}}_{\mathbf{s}}$ as an input.

Model Training

In this work I employ $k = 5$, which is the length of k-mer, $h = 4$, $d_{model} = 256$, and 1024 for the internal dimensionality of the feed-forward network in self-attention encoder.

The loss is mean squared error between the ground truth binding affinity y and predicted value $\hat{y}$. I used the Adam optimizer [17] for training the model parameters and apply dropout [18] to the output of residual connection and self-attention layer for regularization.

Chapter 3

Results

To evaluate the proposed AttendBind method, I reproduced DeepBind and DeeperBind as baselines and trained three models with PBM experiment data from the UniPROBE database [19]. Each PBM data consists of about 40,000 60-base DNA probe and corresponding intensity score representing the relative binding affinity of a given TF to the probes. I chose five TFs (Cbf1, Ceh-22, Oct-1, Rap1, and Zif268) from yeast, worm, mouse and human for evaluation the performance of models and used two PBM array designs for each TF, one for train and validation and the other for test provided by [2].

3.1 Regression Results

To evaluate about 40,000 predicted values for each PBM, I chose spearman's rank correlation coefficient and pearson correlation coefficient between ground truth and predicted values as evaluation metrics.

Table 1 shows the spearman's rank correlation coefficients of the AttendBind

outperformed other models on all five TFs' test data with significant margins, and Table 2 shows same results for pearson correlation coefficients. In both results the DeepBind surpassed the DeeperBind in almost all TFs, but these are quite different from the original DeeperBind paper. In the paper, authors reported the performance of their model as spearman's rank correlation coefficients on two TFs, 0.43 on Ceh-22 PBM data and 0.60 on Oct-1. These numbers are compatible with my implementation of DeeperBind, 0.393 on Ceh-22 and 0.600 on Oct-1. But their implementation of the DeepBind showed significantly poor than mine. On Ceh-22 PBM data, their number is 0.40 and mine is 0.411, and on Oct-1 PBM data, their number is 0.49 but mine is 0.625. Because they used only two TFs and their implementation of the DeepBind is doubtable, they seemed to fail on evaluating the performance of DeepBind correctly.

Table 1: **Spearman's correlation coefficients on 5 TFs.** The AttendBind achieves the best result for whole dataset.

TF	Method		
	DeepBind	DeeperBind	**AttendBind**
Cbf1	0.192	0.152	**0.241**
Ceh-22	0.411	0.393	**0.537**
Oct-1	0.625	0.600	**0.700**
Rap1	0.147	0.156	**0.328**
Zif268	0.479	0.463	**0.494**

Table 2: **Pearson correlation coefficients on 5 TFs.** For whole dataset, the AttendBind outperformed other two models with significant margins.

TF	Method		
	DeepBind	DeeperBind	**AttendBind**
Cbf1	0.529	0.162	**0.821**
Ceh-22	0.474	0.462	**0.762**
Oct-1	0.422	0.42	**0.647**
Rap1	0.074	0.055	**0.284**
Zif268	0.599	0.573	**0.647**

3.2 Binary Classification Results

Additionally, the prediction task can be formulated as binary classification where positive probes are defined as those with actual intensities larger than 4 times standard deviation above the mean of probe intensities in a given experiment array [20]. For each PBM data, I conducted receiver operating characteristic (ROC) curve analysis. Table 3 shows area under the ROC curve (AUC) for each TF indicating the AttendBind outperformed other two baselines, and the ROC curves on five TFs, are depicted in Figure 5.

Table 3: **AUC analysis on 5 TFs.** The AttendBind achieves the best result for all TFs.

TF	Method		
	DeepBind	DeeperBind	**AttendBind**
Cbf1	0.988	0.874	**0.993**
Ceh-22	0.946	0.913	**0.984**
Oct-1	0.946	0.922	**0.97**
Rap1	0.859	0.727	**0.921**
Zif268	0.973	0.952	**0.974**

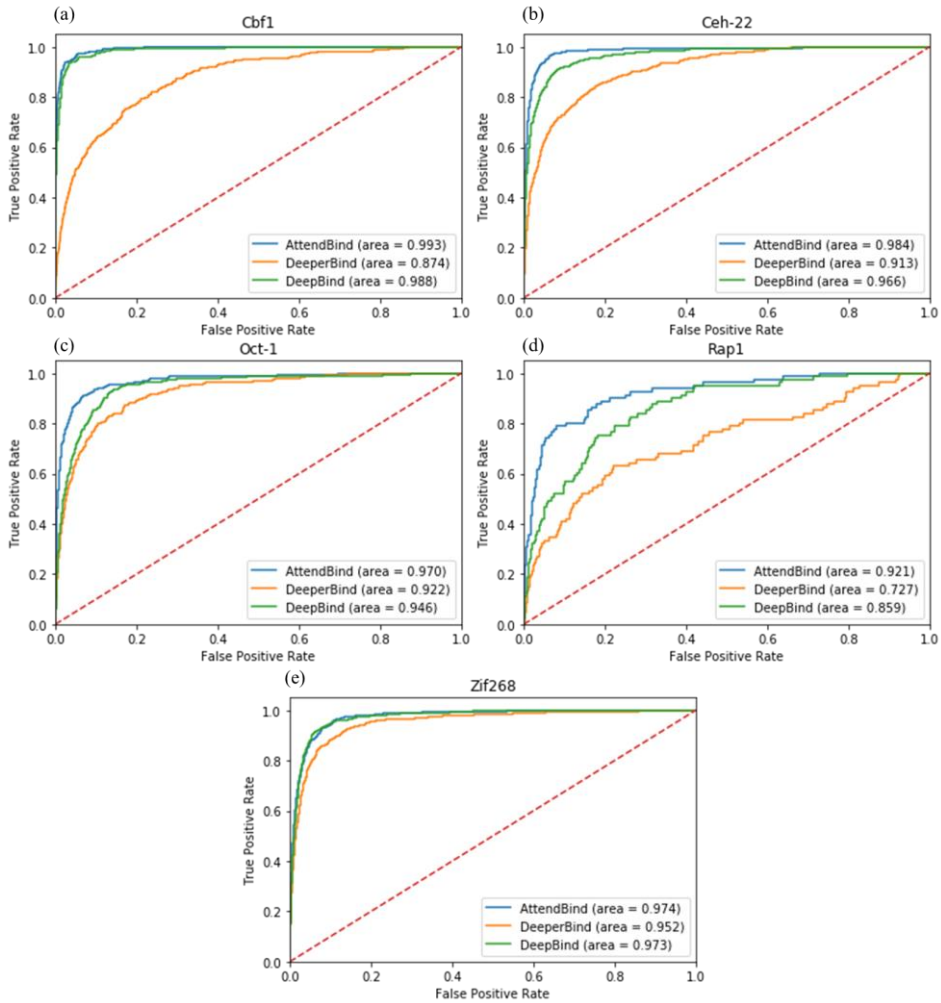

Figure 5 (a)–(e): **Receiver operating characteristic (ROC) curve analysis on 5 TFs. The AttendBind outperformed baselines.**

3.3 Attention Visualization and Motif Analysis

For each $head_i = \text{softmax}\left(\frac{Q_i {K_i}^T}{\sqrt{d_h}}\right) V_i$ in self-attention layer, an attention map A_i is given as,

$$A_i = \text{softmax}\left(\frac{Q_i {K_i}^T}{\sqrt{d_h}}\right) \in \mathbb{R}^{N \times N},$$

where the row vector is summed up to 1, due to the softmax function. In case of column vectors, it can be interpreted as the significance of embedded vectors $x_1, x_2, \ldots, x_N$ within the head. If jth column vector has large value components, x_j thus has large attention weights, and the output $head_i$ reflects x_j more than others. Considering this intuition, it could be possible to determine which k-mer is more important than others by visualizing the attention maps.

Figure 4(c) shows all $h = 4$ attention maps, in transposed form, for the sequence which has the largest predicted affinity in Oct-1 PBM data. Figure 4(a) depicts a single map computed by adding all attention maps and figure 4(b) is known motif (conserved sequence pattern among TF binding sites) of Oct-1 obtained from the UniPROBE database. In figure 4(a), it can be seen that k-mers in the red line box has large attention weights, and nearly equal with the known motif.

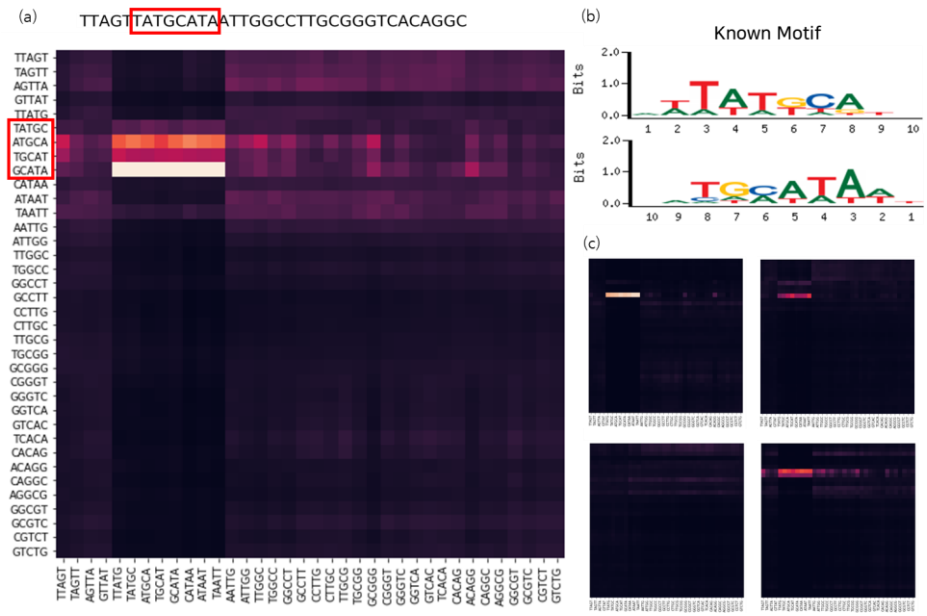

Figure 6 **(c): 4 attention maps (in transposed form) for the sequence "TTAGTTATGCATAATTGGCCTTGCGGTCACAGGC" which has the largest predicted affinity value in Oct-1 PBM data and the known motif of Oct-1. (a): A single attention map computed by adding 4 head attention maps.** The subsequence marked by red line box has large attention weights. **(b): Known motif of Oct-1 from the UniPROBE database.** The two logos illustrate the DNA binding site motif of Oct-1 as graphical representations of the sequence conservation of nucleotides with its information content at each position. The bottom is reverse complement of the upper one.

Agreement with Known Motif

For quantitative analysis, I chose all 418 positive sequences in Oct-1 PBM test data and find some agreement with biologically known motif from the information of the attention maps. In this analysis I define the *Top-3 frequency* of k-mer as follows: for each possible k-mer, where k is 5, the top-3 frequency of a k-mer is initialized as 0 and incremented by 1 when the attention weight of the k-mer has value that is in largest top 3 within a sequence.

Figure 7 shows results of the align agreement of k-mers with known motif of Oct-1. 6 k-mers in Figure 7(a), which has largest top-3 frequencies are aligned in (c). Except the k-mer "GGGGG", all other k-mers are well agreed with known motif in (b). The alignment result indicates that the information from attention maps in self-attention layer is well agreed with known fact, and could provide some useful guideline for scientific discoveries.

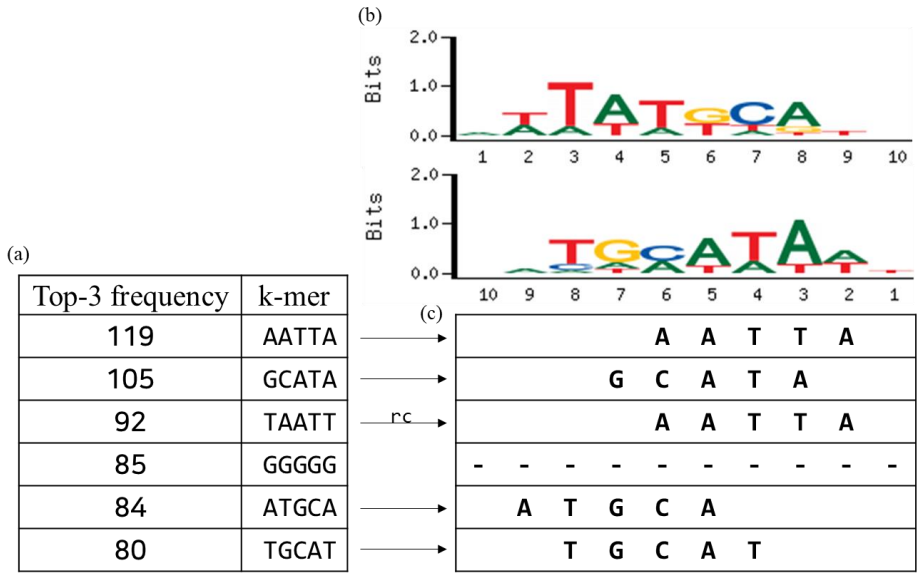

Figure 7: **Top-3 frequencies of k-mers and the result of alignment with known motif of Oct-1.** (a) shows 6 k-mers which has the largest top-3 frequencies. (b) illustrates the motif. (c) is result of sequence alignments of k-mers in (a) with the known motif. The "rc" means reverse complementary convert and the k-mers are well aligned with the known motif (the bottom of (b)) except only "GGGGG".

Motif Analyses on other TFs

To evaluate validity of the motif analysis that I conducted on Oct-1 PBM data as stated earlier, additional analyses were performed in other two TFs, Cbf1 and Zif268.

TF Cbf1 has strongly conserved motif as showed in Figure 8(c). To assess the capability of attention map and top-3 frequency based motif analysis, I listed 5 k-mers which was ranked in top five with top-3 frequency criteria in Figure 8(b). For 177 positive probe sequences in Cbf1 PBM test data, k-mers "CACGT" and "ACGTG" appeared in almost all sequences, and this is explained by the highly conserved sequence "CACGTG" in the motif. In Figure 8(d), it can be seen that all alignments of k-mers are well agreed with the known motif on Cbf1 in (c).

Figure 9 present similar analysis on 1006 positive probes of Zif268 PBM test data. Again, the results of agreement with the motif of Zif268 showed capability of the AttendBind to capture scientifically meaningful pattern from PBM data.

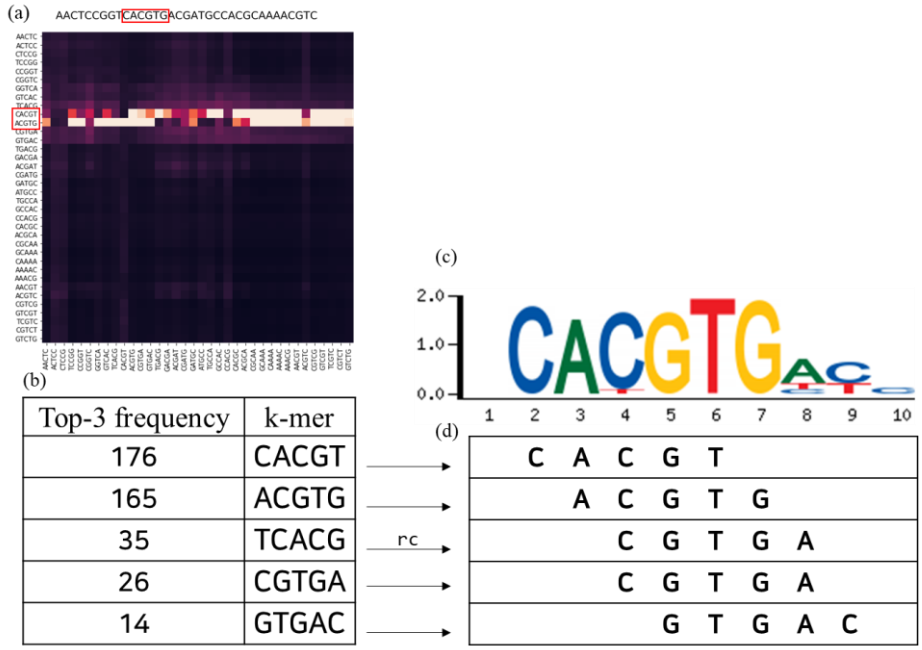

Figure 8: **Attention map based motif analysis on Cbf1 PBM data. (a):** The sum of 4 attention maps for the sequence "AACTCCGGTCACGTGACGATGCC-ACGCAAAACGTC" which has the largest predicted affinity value in the test data. The subsequence "CACGTG" marked by red box has large attention weights and is well agreed with known motif in (c). **(b):** Top 5 k-mers and its top-3 frequencies. **(c):** Known motif of Cbf1 from the UniPROBE database. **(d):** Result of sequence alignments of k-mers in (b) to the known motif. The "rc" means reverse complementary convert.

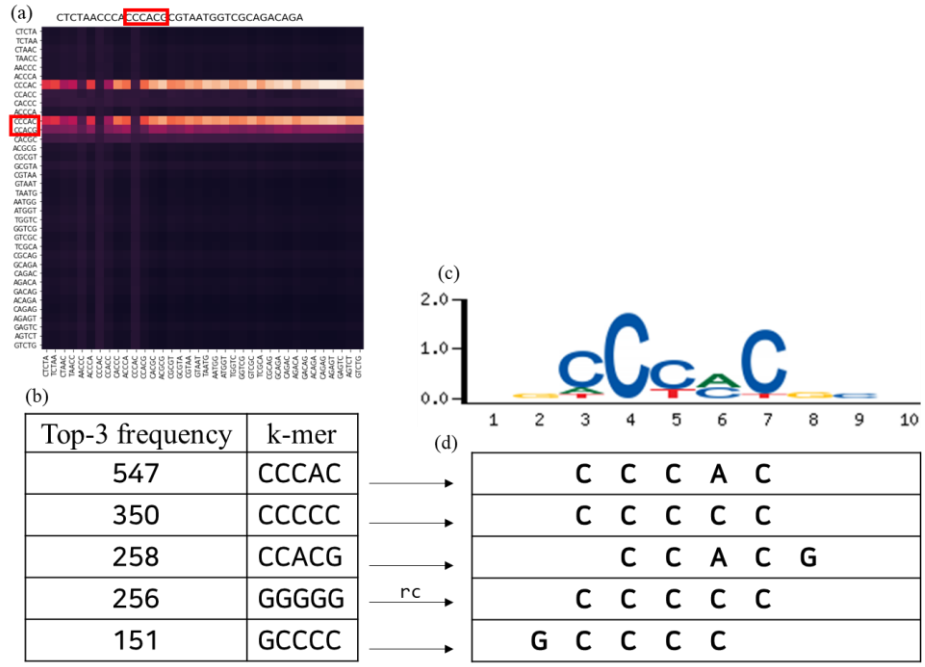

Figure 9: **Attention map based motif analysis on Zif268 PBM data. (a):** The sum of 4 attention maps for the sequence "CTCTAACCCACCCACGCGTAATGG-TCGCAGACAGA" which has the largest predicted affinity value in the test data. The subsequence "CCCACG" marked by red box has large attention weights and is well agreed with known motif in (c). **(b):** Top 5 k-mers and its top-3 frequencies. **(c):** Known motif of Cbf1 from the UniPROBE database. **(d):** Result of sequence alignments of k-mers in (b) to the known motif. The "rc" means reverse complementary convert.

Chapter 4

Conclusion

In this paper I proposed a new approach for predicting TF-DNA binding affinity using the self-attention techniques. Through extensive comparisons with competitive baselines models, it was shown that the new approach for DNA sequence modeling outperforms existing methods. The great success of the AttendBind is due to the self-attention encoder's capability of modeling embedded sequence. Along with the performance gain, attention maps from self-attention layer gave some useful information for interpreting deep learning model and discovering scientific knowledge.

References

[1] Latchman, D. S. Transcription factors: an overview. *The international journal of biochemistry & cell biology* 29(12), 1305-1312 (1997)

[2] Berger, M. F. et al. Compact, universal DNA microarrays to comprehensively determine transcription-factor binding site specificities. *Nature Biotechnology* 24(11), 1429-1435 (2006)

[3] Kharchenko, P. et al. Design and analysis of ChIP-seq experiments for DNA-binding proteins. *Nature Biotechnology* 27, 667-670 (2009)

[4] Lee, D. et al. Discriminative prediction of mammalian enhancers from DNA sequence. *Genome research* (2011)

[5] Warner, J. B. et al. Systematic identification of mammalian regulatory motifs' target genes and functions. *Nature methods* 5(4), 347 (2008)

[6] Alipanahi, B. et al. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature biotechnology* 33(8), 831 (2015)

[7] Hassanzadeh, H. R. & Wang, M. D. DeeperBind: Enhancing prediction of sequence specificities of DNA binding proteins. In *Bioinformatics and Biomedicine (BIBM)* (2016)

[8] Bahdanau, D. et al. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014)

[9] Vaswani, A. et al. Attention is all you need. *Advances in Neural Information Processing Systems* (2017)

[10] Mukherjee, S. et al. Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays. *Nature Genetics* 36(12), 1331-1339 (2004)

[11] Aparicio, O. et al. *Chromatin immunoprecipitation for determining the association of proteins with specific genomic sequences in vivo. Current Protocols in Cell Biology.* Chapter 17. University of Southern California, Los Angeles, California, USA: John Wiley & Sons, Inc. pp. Unit 17.7 (2004)

[12] Wu, Y. et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016)

[13] Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural computation* 9(8), 1735-1780 (1997)

[14] Cho, K. et al. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR, abs/1412.3555* (2014)

[15] He, K. et al. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778 (2016)

[16] Ba, J. L. et al. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016)

[17] Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014)

[18] Srivastava, N. et al. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15(1), 1929–1958 (2014)

[19] Newburger, D. E. & Bulyk, M. L. UniPROBE: an online database of protein binding microarray data on protein–DNA interactions. *Nucleic Acids Research* 37(Database issue), D77–D82 (2009)

[20] Weirauch, M. T. et al. Evaluation of methods for modeling transcription factor sequence specificity. *Nature biotechnology* 31(2), 126 (2013)

요약

전사인자는 DNA 프로모터에 결합하여 전사를 개시하는 단백질 집합으로, 유전자 발현 및 조절 과정에서 매우 중요한 요소를 차지한다. 전사인자는 DNA 서열 특이성을 가지며 이를 예측하기 위한 모델이 꾸준히 제시되었다. 본 연구에서는 전사인자-DNA 결합 친화도를 예측하기 위해 k-mer 임베딩 및 자기참조(self-attention) 기반의 딥러닝 모델을 만들어 단백질 결합 마이크로어레이(PBM) 데이터를 이용하여 학습 및 성능 평가를 실시하였다. 실험을 통해서 본 모델이 컨볼루션 신경망(Convolutional Neural Network)과 순환 신경망(Recurrent Neural Network) 기반의 경쟁 모델을 큰 차이로 앞서는 결과를 얻었고 이를 통해 k-mer 임베딩과 자기참조 모델링이 DNA 서열을 다루는 데에 좋은 방법론임을 밝혔다. 그리고 자기참조 모델링을 통해 얻을 수 있는 어텐션 지도(attention map)를 통해 과학적으로 의미 있는 지식을 발견할 수 있다는 것도 보일 수 있었다.

감사의 글

2년여의 연구실 생활 동안, 매 순간 많은 분들의 도움을 받았습니다. 가장 먼저 여러모로 성숙하지 못한 저를 가르쳐 주신 김선 교수님께 감사드립니다. 교수님의 생각과 말과 행동은 앞으로도 저에게 큰 귀감이 되어줄 것입니다. 고맙습니다.

대학원 생활을 물심양면으로 지원해주신 박근수 교수님께 감사의 말씀을 드립니다. 이끌어 주신 덕분에 어려움을 이겨내고 대학원 생활을 마칠 수 있었습니다. 고맙습니다.

연구실 동료 분들께 감사드립니다. 다시 없을 만큼, 분에 넘치게 좋은 친구들을 사귈 수 있었습니다. 덕분에 나날이 즐겁고 행복했습니다. 든 자리는 몰라도 난 자리는 안다는데 제가 떠난 자리는 금세 더 좋은 것으로 채워지기를 바랍니다. 고맙습니다.

그리고 엄마, 아빠, 동생에게 글로는 채 전하지 못하는 고마운 마음을 살면서 평생 전하겠습니다.

감사합니다.