IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 March 2024, accepted 12 May 2024, date of publication 15 May 2024, date of current version 22 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3401464

RESEARCH ARTICLE

Needleman-Wunsch Attention: A Framework for Enhancing DNA Sequence Embedding

KYELIM LEE[1] AND ALBERT NO[2], (Member, IEEE)
[1]Department of Electronic and Electrical Engineering, Hongik University, Seoul 04066, South Korea
[2]Department of Artificial Intelligence, Yonsei University, Seoul 03722, South Korea

Corresponding author: Albert No (albertno@yonsei.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) grant funded by Korean Government [Ministry of Science, ICT and Future Planning (MSIP)] under Grant NRF-2022M3C1A3081366.

ABSTRACT In many biological research studies that rely on DNA sequence data, calculating the edit distance between two sequences is a vital component. However, computing the edit distance involves dynamic programming, which can be computationally intensive. To address this challenge, numerous works have focused on embedding sequences into the vector space while preserving the distance metric. This means that the edit distance between sequences is analogous to the distance between their corresponding vectors. In this study, we propose a novel Needleman-Wunsch Attention (NWA) framework for sequence embedding that leverages the relationship between the Needleman-Wunsch (NW) matrix and attention maps to improve the accuracy and efficiency of edit distance approximation methods. Our approach applies to any deep learning-based sequence embedding network and provides a general solution to improve the accuracy and efficiency of edit distance approximation methods. We validate the effectiveness of our proposed method by applying it to various existing embedding networks, demonstrating improved edit distance-preserving embedding in an actual dataset. The code is publicly available at https://github.com/thisislim/nw-attention/.

INDEX TERMS Attention, edit distance, DNA sequence, Needleman-Wunsch, sequence embedding.

I. INTRODUCTION

The exponential growth of DNA sequencing technologies has resulted in an extensive accumulation of DNA data, making the analysis of DNA sequences a daunting task. Consequently, there has been a surge of interest in data-driven bioinformatics research [1]. The edit distance [2], also known as the Levenshtein distance [3], is a critical tool in bioinformatics, allowing researchers to quantify the minimum number of edit operations necessary to transform one string into another. This distance metric is invaluable in numerous bioinformatics tasks, including sequence clustering and multiple sequence alignment (MSA). For instance, in DNA storage, sequences are read multiple times, and clustering the sequencing results can enhance the quality of the reading procedure [4], [5]. Furthermore, phylogenetics [6], a field of study concerned with evolutionary relationships between species, utilizes the edit distance to compare similarities and differences in DNA, RNA, or protein sequences between different species and infer their evolutionary relationships. To represent the phylogenetic tree [7], hierarchical clustering [8] is necessary, requiring an edit distance computation [9], [10].

Although edit distance is a useful metric for comparing DNA sequences, computing it using dynamic programming algorithms [11], [12] such as Needleman-Wunsch (NW) [13] can be computationally expensive. The NW algorithm generates a similarity matrix based on gap-costs, finding the optimal path with minimal cost that traces back to the origin. However, constructing the NW matrix has a quadratic computational cost with respect to sequence length, making it challenging to conduct large-scale studies with the growing amount of encoded DNA sequences. To address this issue, alignment-free methods have been introduced to approximate the edit distance between biological sequences. These approaches offer low complexity by circumventing

The associate editor coordinating the review of this manuscript and approving it for publication was Abdullah Iliyasu.

the need for explicit alignment [14], [15]. For example, FFP [15] compares two sequences using Jensen-Shannon divergence [16] between distributions of k-mers.

In recent years, deep learning-based edit distance embedding networks have been proposed as an alternative to approximation techniques. The aim is to create an embedding from DNA sequences to vectors that conserve the distance structure. Essentially, the embedding (the neural network) is trained to ensure that the edit distance between any two DNA sequences corresponds to the distance between their respective vector representations [4], [17], [18], [19], [20]. Specifically, CNN-ED [19] uses a convolutional neural network (CNN) to embed the DNA sequence, while NeuroSEED [18] utilizes both CNN and Transformer-based networks trained to match hyperbolic distance metrics between vectors [21], [22]. DSEE [4] has the same network architecture as CNN-ED, but trains the model with an additional constraint to match the χ^2 distribution.

The deep learning methods mentioned earlier offer an intriguing alternative to traditional dynamic programming-based edit distance techniques. However, their main emphasis is on a loss function that either evaluates scalar output (such as hyperbolic distance) or gauges the distributional distance based on the assumption of a chi-squared distribution with a singular parameter. These methods consistently depend on the first-order statistics of both the target and the trained distribution of distances. However, to maintain the edit distance between any sequence pair, the embedded vectors should incorporate alignment information. To fully understand the relationships between arbitrary DNA sequence pairs, it is beneficial to enrich the embedding with supplementary information that better captures the edit distance between the sequences.

In this work, we propose the Needleman-Wunsch Attention (NWA) framework, which informs the alignment structure more directly to the embedding network. The primary driving force behind our research is the similarity between the attention map and the Needleman-Wunsch (NW) matrix. The attention map identifies the corresponding elements of two sequences, whereas the NW matrix represents the alignment structure between these sequences. Therefore, our central concept involves a direct comparison of the attention map of two embedded vectors and the NW matrix of the input sequences. Specifically, for a given deep learning-based encoder that embeds a pair of DNA sequences to embedded vectors, we propose an additional regularization term with an additional cross-attention module. The cross-attention module generates an attention map of input sequence pairs that describes the correspondence between bases of input sequences. Then, we can add a regularizer term by measuring the difference between the NW matrix of input sequences and the attention map obtained from the cross-attention module. It is important to note that the NWA framework is not a complete replacement for existing deep learning-based edit distance embedding models but rather a complementary addition. This feature makes NWA applicable to any existing deep learning-based method, providing a way to improve alignment accuracy without fundamentally altering the underlying model architecture.

To evaluate the effectiveness of the proposed NWA framework, we applied it to existing schemes, including variations of NeuroSEED [18] and DSEE [4], using an actual dataset. Our experimental results demonstrate that embedding models trained using NWA achieve improved edit distance-preserving mapping compared to those trained without NWA.

A summary of our contributions is as follows:

- We propose the NWA framework, which leverages the relationship between the Needleman-Wunsch (NW) matrix and attention maps to improve sequence embedding.
- Our approach applies to any deep learning-based sequence embedding neural networks, providing a general solution to improve the accuracy and efficiency of edit distance approximation methods.
- We validate the effectiveness of our proposed method by applying it to existing embedding networks, demonstrating an improved edit distance-preserving mapping in an actual dataset.

II. RELATED WORKS

In this section, we briefly review related works, especially the Needleman-Wunsch algorithm, attention mechanism, and sequence embedding for edit distance.

A. NEEDLEMAN-WUNSCH ALGORITHM

The Needleman-Wunsch (NW) algorithm [13] is a classical dynamic programming-based method for sequence alignment. Given two sequences $\boldsymbol{s}^{(1)} = (s_1^{(1)}, \ldots, s_{n_1}^{(1)})$ and $\boldsymbol{s}^{(2)} = (s_1^{(2)}, \ldots, s_{n_2}^{(2)})$ with lengths n_1 and n_2, respectively, the NW algorithm generates a $(n_1 + 1) \times (n_2 + 1)$ matrix M with entries from $M_{0,0}$ to M_{n_1,n_2}, which is filled from the origin $M_{0,0} = 0$. Each cell score is calculated by choosing the maximum scores derived from previous cells. The score candidates are obtained by adding the substitution score to the top-left cell and the gap cost c_{gap} to the left and top cells, depending on the insertion, deletion and substitution errors. Specifically, the value in cell (i, j) of the matrix is calculated as

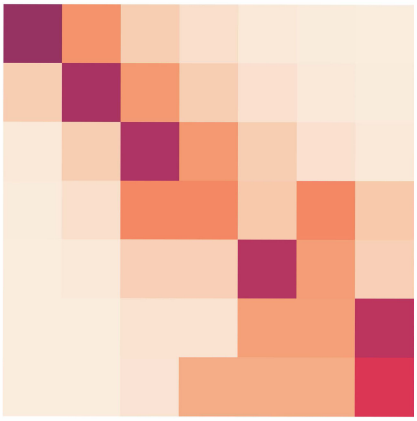
$$M_{i,j} = \max \begin{cases} M_{i-1,j-1} + c_{sub}(s_i^{(1)}, s_j^{(2)}) \\ M_{i-1,j} + c_{gap} \\ M_{i,j-1} + c_{gap} \end{cases} \tag{1}$$

where $c_{sub}(s_i^{(1)}, s_j^{(2)})$ indicates the substitution score between $s_i^{(1)}$ and $s_j^{(2)}$.

Once the NW matrix is built, the algorithm aligns the sequences by tracing back to the origin from the bottom-right cell of the matrix M_{n_1,n_2}. The NW matrix shows viable alignment paths, and the algorithm returns high-quality alignments. Despite its high computational cost, the NW algorithm is still widely used in bioinformatics, and we

		G	T	T	C	A	G
	0	-1	-2	-3	-4	-5	-6
G	-1	1	0	-1	-2	-3	-4
T	-2	0	2	1	0	-1	-2
A	-3	-1	1	1	0	1	0
C	-4	-2	0	0	2	1	0
G	-5	-3	-1	-1	1	1	2
T	-6	-4	-2	0	0	0	1

(a) Needleman-Wunsch matrix



(b) Normalized Needleman-Wunsch matrix

FIGURE 1. An example of needleman-wunsch (NW) matrix.

can directly derive an edit distance between two sequences from the NW matrix. An example of an NW matrix of the Needleman-Wunsch algorithm, when $\boldsymbol{s}^{(1)} = \texttt{GTACGT}$ and $\boldsymbol{s}^{(2)} = \texttt{GTTCAG}$, is shown in Fig. 1(a). Fig. 1(b) presents the normalized NW matrix where we apply the softmax operation to each row.

B. ATTENTION

Transformer architecture-based models, which are dominated by the attention mechanism, have become prevalent in various fields such as natural language processing and computer vision. Language models (LM) [23], [24], [25], [26], [27] based on the Transformer architecture have significantly improved the accuracy of natural language processing tasks. In computer vision, Transformer-based models have also demonstrated state-of-the-art performance [28], [29], [30], [31]. For each component of the sequence, an attention module [32], [33], [34] finds the most relevant component in a given sequence. Self-attention seeks relevance within a single input sequence, while cross-attention correlates to two input sequences. Specifically, each component has a query, key, and value triplet (q, k, v). The attention module finds the matching key for a given query and obtains the corresponding value. The soft attention [32] is calculated as a soft-maxed dot product of q and k, which is $\texttt{softmax}\left(\frac{qk^\intercal}{\sqrt{d_k}}\right)v$.

Multi-head attention allows each component of the sequence to have multiple triplets of (q, k, v) to obtain various aspects of correlations. The outputs of multiple heads are concatenated and linearly projected to produce the final output.

The attention map is a byproduct of the attention mechanism, which shows the weight (an inner product of key and query) of each component mapped by a neural network. Suppose that the Transformer encoder and decoder pair processes a sequence $\boldsymbol{s}^{(1)}$ and $\boldsymbol{s}^{(2)}$ with lengths n_1 and n_2, respectively. If we add a <sos> token (start of sequence) to both sequences, the attention map is also an $(n_1+1)\times(n_2+1)$ matrix, where an (i, j)-th component of the attention map indicates the relevance between the i-th component of $s_i^{(1)}$ and the j-th component $s_j^{(2)}$. Hence, it is natural to associate the attention map with the NW matrix, as the NW matrix illustrates the aligned components of the corresponding sequences. The connection between the NW matrix and the attention map is a key component of our proposed NWA framework, which we will discuss in detail in the following sections.

C. SEQUENCE EMBEDDING FOR EDIT DISTANCE

DNA sequence embedding maps DNA sequences into vector representations. The primary objective is to ensure that the vector distance approximates the edit distance of the corresponding input sequences. This approximation allows for an efficient estimation of the edit distance between sequence pairs through vector distance calculations. There are numerous methods for sequence embedding, including k-mers frequency analysis [14], [35] that do not rely on deep learning. A k-mer refers to a subsequence of k nucleotides within a DNA sequence, which provides a representation of the DNA sequence. This representation can be viewed as a DNA sequence embedding.

A series of studies have employed deep learning-based techniques to train distance-preserving sequence embeddings. Specifically, these methods aim to train an encoder (embedding network f_θ) so that the distance between the vectors $\boldsymbol{z}(1) = f_\theta(\boldsymbol{s}(1))$ and $\boldsymbol{z}(2) = f_\theta(\boldsymbol{s}(2))$ closely mirrors the edit distance between the inputs $\boldsymbol{s}(1)$ and $\boldsymbol{s}(2)$. EINN [36] proposed a differential sequence alignment algorithm that replaces the maximum operation of the NW algorithm with softmax, however, the complexity was still too high. SENSE [20] was the first attempt to use deep learning for alignment-free sequence analysis, which uses the Siamese neural network to learn embeddings. Dai et al. [19] proposed CNN-ED, where the embedding network is a convolutional neural network and trained with triplet loss [37]. Corso et al. [18] proposed the NeuroSEED framework where the authors considered a hyperbolic distance as a distance metric of the embedding vector space. The NeuroSEED embedding network can utilize a variety of structures, such as CNN, GRU, and Transformer. The authors demonstrated

the efficacy of training these structures under hyperbolic distance for Hierarchical Clustering (HC) [38] and Multiple Sequence Alignment (MSA) [39]. Similarly, DSEE [4] trained embedding networks of diverse architectures, such as CNN, RNN, and GRU [40], using a chi-squared regression to align the distribution of sequence distances with the distribution of vector distances. Chen et al. [17] introduced AsMac, a neural network model designed for approximate string matching. This model learns patterns from the data to predict the alignment distance of given sequence pairs.

In the following sections, we will introduce our proposed NWA framework for sequence embedding, which is applicable to any deep learning-based edit distance embedding model. We will show that our method improves the edit distance-preserving mapping of existing embedding models.

III. PROBLEM FORMULATION

In this study, we focus on DNA sequences of fixed length $n \geq 1$ consisting of nucleotides represented by $\mathcal{S} = \{A, C, G, T\}$. A sequence is denoted as $\boldsymbol{s} = (s_1, s_2, \ldots, s_n)$ where $s_i \in \mathcal{S}$ for $1 \leq i \leq n$. An encoder, $f : \mathcal{S}^n \to \mathbb{R}^m$, transforms these DNA sequences into vectors of dimension m, where $m \geq 1$ denotes the embedding dimension. Let $ED : \mathcal{S}^n \times \mathcal{S}^n \to [0, \infty)$ refer to an edit distance measure between sequences of length n, while $d : \mathbb{R}^m \times \mathbb{R}^m \to [0, \infty)$ is a distance measure between m-dimensional vectors. We have not limited ourselves to the mean squared error for the distance measure d.

We aim to introduce a universal framework in this research that enhances an existing deep-learning-based embedding mechanism. To be more specific, let f_θ be a neural network characterized by the parameter θ, which converts a sequence into an embedded vector. The model is trained to reduce the encoder loss $\mathcal{L}_{enc}(\theta)$. An instance of such an encoder loss could be an edit distance loss, that is, $\mathcal{L}_{enc}(\theta) = \mathcal{L}_{ed}(\theta)$ where

$$\mathcal{L}_{ed}(\theta) = \sum_{\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}} \left(ED(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}) - d(f_\theta(\boldsymbol{s}^{(1)}), f_\theta(\boldsymbol{s}^{(2)})) \right)^2. \tag{2}$$

Note that the summation is over given pairs of sequences $\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)} \in \mathcal{S}^n$. For this given network f_θ and the corresponding loss function $\mathcal{L}_{enc}$, we propose an additional regularization term $\mathcal{L}_{reg}(\theta)$ to improve the training procedure. The regularizer is based on the relation between the Needleman-Wunsch matrix and attention map, which we provide in detail in Section IV.

A. PERFORMANCE METRIC

Following the convention from NeuroSEED [18], we evaluate the performance of the trained embedding network with root mean squared error (RMSE) between the edit distance and the vector distance:

$$\text{RMSE} = \frac{100}{n} \sqrt{\sum_{\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}} \left(ED(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)}) - d(\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}) \right)^2} \tag{3}$$

where $\boldsymbol{z}^{(1)} = f_\theta(\boldsymbol{s}^{(1)})$ and $\boldsymbol{z}^{(2)} = f_\theta(\boldsymbol{s}^{(2)})$.

In certain applications, such as clustering in DNA storage, approximating smaller edit distances is of greater importance. In order to focus on sequence pairs that are closely related, we define a pair of sequences as K-homologous if the edit distance between them is less than or equal to K. We then measure the root mean square error (RMSE) of these K-homologous pairs. Typically, we set K to 40, following the methodology established by Guo et al. [4].

IV. NEEDLEMAN-WUNSCH ATTENTION (NWA)

In this section, we present our proposed framework, Needleman-Wunsch Attention (NWA), which facilitates learning the sequence alignment structure using Needleman-Wunsch matrices.

A. MOTIVATION

Consider an attention map between two sequences of vectors. Given the query and the key vector $q, v \in \mathbb{R}^{n \times h}$, the attention map calculates the dot product $\boldsymbol{a} = qk^\intercal$ (this is considered prior to the multiplication by "value"). The resulting attention map $\boldsymbol{a} \in \mathbb{R}^{n \times n}$ shows pronounced activations where the component $\boldsymbol{a}_{i,j}$ indicates a correlation between the i-th component of $\mathbf{u}$ and the j-th component of $\mathbf{v}$; that is, between $\mathbf{u}_i$ and $\mathbf{v}_j$.

Similarly, the NW matrix, as shown in Fig. 1(a), highlights an alignment path between the input sequences $\boldsymbol{s}^{(1)}$ and $\boldsymbol{s}^{(2)}$ in $\mathcal{S}^n$. As demonstrated in Fig. 1(b), if the (i, j)-th component of the NW matrix is large, it suggests that $\boldsymbol{s}_i^{(1)}$ and $\boldsymbol{s}_j^{(2)}$ are likely to match in the optimal alignment of the two sequences. This behavior mirrors that of the attention map when applied to length-n sequences of vectors, $\mathbf{u}, \mathbf{v} \in \mathbb{R}^{n \times h}$.

Observing the parallels between the NW matrix and the attention map, it is intuitive to design an attention map–produced by the neural network–to capture the nuances of the alignment distribution. This concept underpins the foundation of our work, as mentioned previously. By incorporating NWA, we anticipate the encoded vectors to capture not just edit distance information, but also nuanced alignment patterns between the analyzed sequences. Through the introduction of NWA, we expect the encoded vectors to encapsulate not only the edit distance information, but also the intricate alignment relationships between the sequences in question.

B. NEEDLEMAN-WUSNCH ATTENTION FRAMEWORK

We assume that an embedding neural network (encoder) f_θ and a loss function $\mathcal{L}_{enc}(\theta)$ are provided. Our objective is to improve such training procedures by introducing an additional regularizer term that supports the training process. We propose the *Needleman-Wunsch Attention (NWA)* method, which directly integrates alignment information into the training process. NWA includes an additional decoder network that produces an attention map, effectively illustrating the correlation between the components of the two input sequences. By utilizing this approach, the embedding network (encoder) can learn the alignment

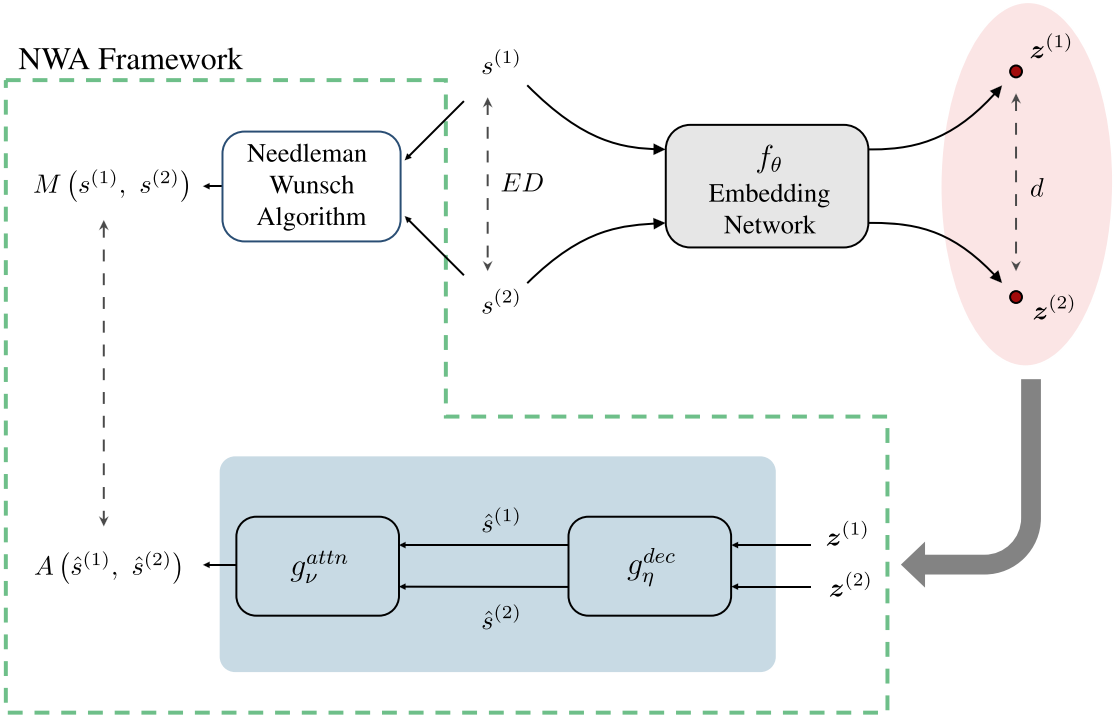


FIGURE 2. An illustration of needleman-wunsch attention (NWA) framework.

structure of sequences, further enhancing its performance and accuracy.

For given sequences $\boldsymbol{s}^{(1)}$ and $\boldsymbol{s}^{(2)}$ in $\mathcal{S}^n$, the encoder produces embedded vectors $\boldsymbol{z}^{(1)}$ and $\boldsymbol{z}^{(2)}$ in $\mathbb{R}^m$. We introduce the decoder network, denoted as $g_\eta^{dec} : \mathbb{R}^m \to \mathbb{R}^{n\times h}$. This process can be interpreted as reconstructing the input sequences into a sequence of vectors with a length of n. In other words, $\hat{\boldsymbol{s}}^{(1)} = g_\eta^{dec}(\boldsymbol{z}^{(1)})$ and $\hat{\boldsymbol{s}}^{(2)} = g_\eta^{dec}(\boldsymbol{z}^{(2)})$, both in $\mathbb{R}^{n\times h}$. Our objective is to train the encoder f_θ in such a way that the decoder can effectively reconstruct $\hat{\boldsymbol{s}}^{(1)}$ and $\hat{\boldsymbol{s}}^{(2)}$, ensuring that they retain alignment information. To assess if these reconstructions contain the desired alignment information, we produce an attention map between the two sequences of vectors using another network g_ν^{attn}, then compare it with the NW matrix.

The proposed decoder g_η^{dec}, similar to the Transformer encoder, consists of multiple layers of multihead self-attention modules and a feedforward network. The cross-attention module g_ν^{attn}, which generates attention map, takes the outputs of decoder $g_\eta^{dec}(\boldsymbol{z}^{(1)})$ and $g_\eta^{dec}(\boldsymbol{z}^{(2)})$, and generates $n \times n$ attention map $A(\hat{\boldsymbol{s}}^{(1)}, \hat{\boldsymbol{s}}^{(2)})$.

Note that g_ν^{attn} computes a dot product between the outputs of g_η^{dec}, followed by a linear layer that merges weighted sums from multiple heads. The final layer of g_ν^{attn} is a softmax layer, ensuring that each row of the attention matrix sums to 1. In summary, we designate $g_{\eta,\nu} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^{n\times n}$ as an attention map generating network, a simple composition of the decoder and cross-attention module, which is depicted in Fig. 3.

$$g_{\eta,\nu}(\boldsymbol{z}^{(1)}, \boldsymbol{z}^{(2)}) = g_\nu^{attn}(g_\eta^{dec}(\boldsymbol{z}^{(1)}), g_\eta^{dec}(\boldsymbol{z}^{(2)})). \tag{4}$$

On the other hand, let $M_{NW}(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)})$ represent a Needleman-Wunsch (NW) matrix of sequences $\boldsymbol{s}^{(1)}$ and $\boldsymbol{s}^{(2)}$. As the NW matrix presents feasible alignment paths, we aim to train the attention map generating network $g_{\eta,\nu}$ to produce an attention map resembling the NW matrix. We apply softmax to normalize the NW matrix, as described in Fig. 1(b). To compare the NW matrix and attention map during training, we introduce an additional loss function

$$\mathcal{L}_{nw}(\theta, \eta, \nu) = \sum_{\boldsymbol{s}^{(1)},\boldsymbol{s}^{(2)}} JSD\left(g_{\eta,\nu}(f_\theta(\boldsymbol{s}^{(1)}), f_\theta(\boldsymbol{s}^{(2)})), M_{NW}(\boldsymbol{s}^{(1)}, \boldsymbol{s}^{(2)})\right) \tag{5}$$

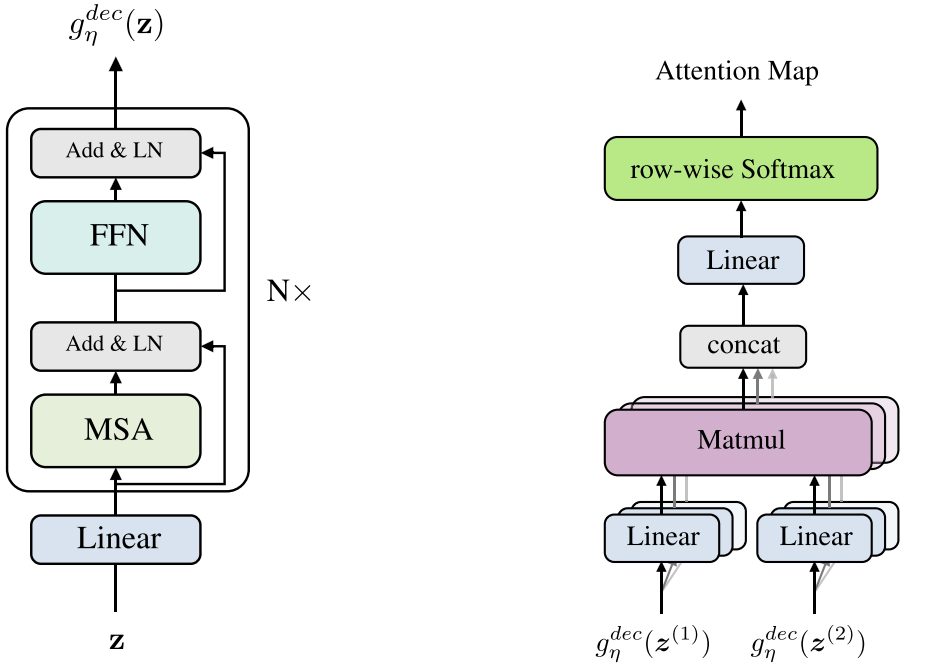
where $JSD(\cdot,\cdot)$ computes the sum of row-wise Jensen-Shannon divergences (JSD) [16], [41]. JSD is a commonly utilized metric between probability measures, making it suitable in our context since rows of both the attention map and NW matrix are outputs of the softmax layer and can be perceived as a probability distribution.

Finally, the total loss is defined as

$$\mathcal{L}_{total}(\theta, \eta, \lambda) = \mathcal{L}_{enc}(\theta) + \lambda\mathcal{L}_{nw}(\theta, \eta, \nu), \tag{6}$$

where $\lambda > 0$ is an adjustable parameter and the proposed loss $\mathcal{L}_{nw}$ functions acts as a regularizer. With the added support of $\mathcal{L}_{nw}$, the encoder can more effectively learn the hidden structure of the sequences.

In practical applications where a DNA sequence is embedded in the vector space, only the encoder f_θ is required. The role of decoder networks is primarily to facilitate the training of a more effective encoder.



(a) Decoder g_η^{dec} (b) Cross Attention g_ν^{attn}

FIGURE 3. Decoder and cross-attention architectures for the needleman-wunsch attention (NWA) framework.

V. EXPERIMENTS

In this section, we present the experimental details of the proposed methodology for a variety of embedding networks, using actual datasets. Detailed experimental setups including network architectures as well as training parameters are provided.

A. DATASETS

For the model performance evaluation, we used the Qiita [42] and DNA-Fountain [43] dataset. The Qiita dataset comprises human microbiome samples and their associated metadata. Each sequence in the Qiita dataset has a length of 152. DNA-Fountain sequence data consist of generated references and reads from their DNA Storage experiments. DSEE [4] sampled homologous and non-homologous sequences from DNA-Fountain sequences at a 1:1 ratio. On the other hand, we sampled oligos for training from a set of sequences in a way similar to the NeuroSEED setup.

The sequences were sampled and divided into training, validation and test sets, containing 7,000, 700, and 1500 sequences, respectively. These sets were further subdivided to facilitate the calculation of the edit distance and Needleman-Wunsch (NW) matrix. Each subdivision contained 350 sequences, maintaining edit distance and NW matrices for 350^2 pairs in the training and validation data. In contrast, the subdivision count for the test set was 300.

During the preprocessing phase, we pre-calculate both the ground truth edit distances and NW matrices. Since the NW matrix $M(s_1, s_2)$ is the $(n+1) \times (n+1)$ matrix, it is necessary to adjust the dimensions of the attention and NW matrices. To rectify this mismatch, a <sos> token is padded at the beginning of each sequence. Consistent with the input protocol of the vision transformer [28], no <eos> token is padded at the end of the sequence. To simplify notation, we refer to the length of input sequences, including padding, as L, where $L = n + 1$. As a result, a dataset is assembled from sequence pairs (s_1, s_2), their associated edit distances, and the corresponding NW matrices.

B. EMBEDDING NETWORKS

NWA does not introduce a new encoder framework; rather, it supplements existing encoders with additional mechanisms to enhance the transfer of alignment information. Accordingly, the embedding networks used in our experiments adhere to the representative configurations of the original papers. In this section, we briefly discuss the encoder networks utilized in our experiments.

1) NEUROSEED

Klimovskaia et al. corso2021neural claimed that the hyperbolic distance captures the implicit hierarchical structure of biological evolution [22], resulting in a reduction in the root mean square error (RMSE). To calculate the hyperbolic distance, the NeuroSEED encoder applies a Poincarè projection to the output, confining it to a unit ball $\mathcal{B}_m = \{\boldsymbol{x} : \|\boldsymbol{x}\| \leq 1\}$, i.e., $f : \mathcal{S}^n \to \mathcal{B}_m$. For $z_1, z_2 \in \mathcal{B}_m$, the hyperbolic distance between the two embeddings is defined by

$$d(z_1, z_2) = \operatorname{arcosh}\left(1 + 2\frac{||z_1 - z_2||^2}{(1 - ||z_1||^2)(1 - ||z_2||^2)}\right), \quad (7)$$

where $\|\cdot\|$ represents the conventional Euclidean norm. Subsequently, the encoding loss is simply an edit distance loss, as delineated in (2).

NeuroSEED evaluates a variety of embedding networks. However, this study focuses primarily on NeuroSEED-CNN and NeuroSEED-Transformer (referred to in shorthand as CNN and Transformer), adhering primarily to the original NeuroSEED experimental setup. Nevertheless, to enhance the performance of the embedding, we increased the number of channels from 32 to 64, while retaining the number of layers at 4. The Global Transformer from NeuroSEED encodes sequences without any mask, whereas the Local Transformer attends to each nucleotide on either side of the current position. In our Transformer experiment, we deployed the global Transformer. We also set a segment size of 4 as described in the NeuroSEED setup, which implies that the encoder processes 4 bases as a single word. While the NeuroSEED Transformer utilizes sinusoidal positional encoding, instead, we incorporate learnable positional encoding [44], [45], constructing each token from 4 or 8 bases. When applied to the Qiita dataset, our 2-layer Transformer encoder features 2 attention heads and sets all hidden dimensions (i.e. d_{model}, $d_{feedforward}$) to 16.

2) DSEE

Guo et al. [4] trained embedding neural network using chi-squared regression. The idea is that the distribution of the edit distance between the randomly chosen sequence pairs should follow the chi-squared distribution. Thus, DSEE is trained to minimize the KL divergence [46] between the edit distance distribution and the distance distribution of vector pairs. For a given batch of sequence pairs, DSEE computes the mean of vector distances x and the mean of edit distances y. Then, the corresponding loss function would be

$$\mathcal{L}_{ed}(\theta) = \frac{y}{2} + \log\Gamma\left(\frac{y}{2}\right) - \left(\frac{y}{2} - 1\right)\log x + \frac{x}{2}\log e \quad (8)$$

where the Γ is the Gamma function. Note that (8) is a KL divergence between two chi-squared distribution with degrees of freedom x and y.

DSEE integrated CNN-ED [19] as their CNN encoder, yielding an RMSE of 6.17 ± 0.24 for the Qiita dataset in our run. Consequently, we adopted the NeuroSEED CNN architecture for chi-squared regression. Although we utilized a different model, the foundational concept of DSEE – applying chi-squared regression to various encoders – remains intact. We also made a concerted effort to adhere as closely as possible to the original configuration, maintaining 64 channels and 5 layers, as delineated in the initial paper.

C. TRAINING PARAMETERS

For NWA to successfully capture distance structure and alignment distribution, it is critical to properly scale the value of λ, a pivotal parameter of our model. For a comprehensive understanding of the impact of λ, Fig. 4 illustrates the RMSE of the λ scaling for the Qiita dataset with

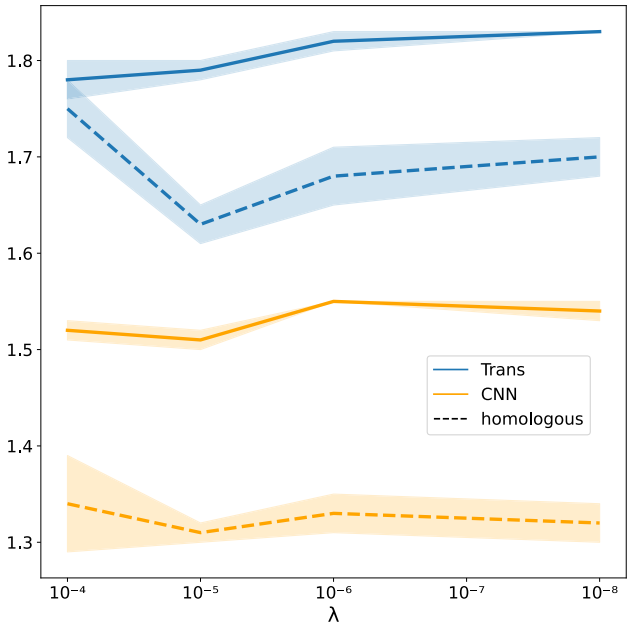


FIGURE 4. λ scaling curve.

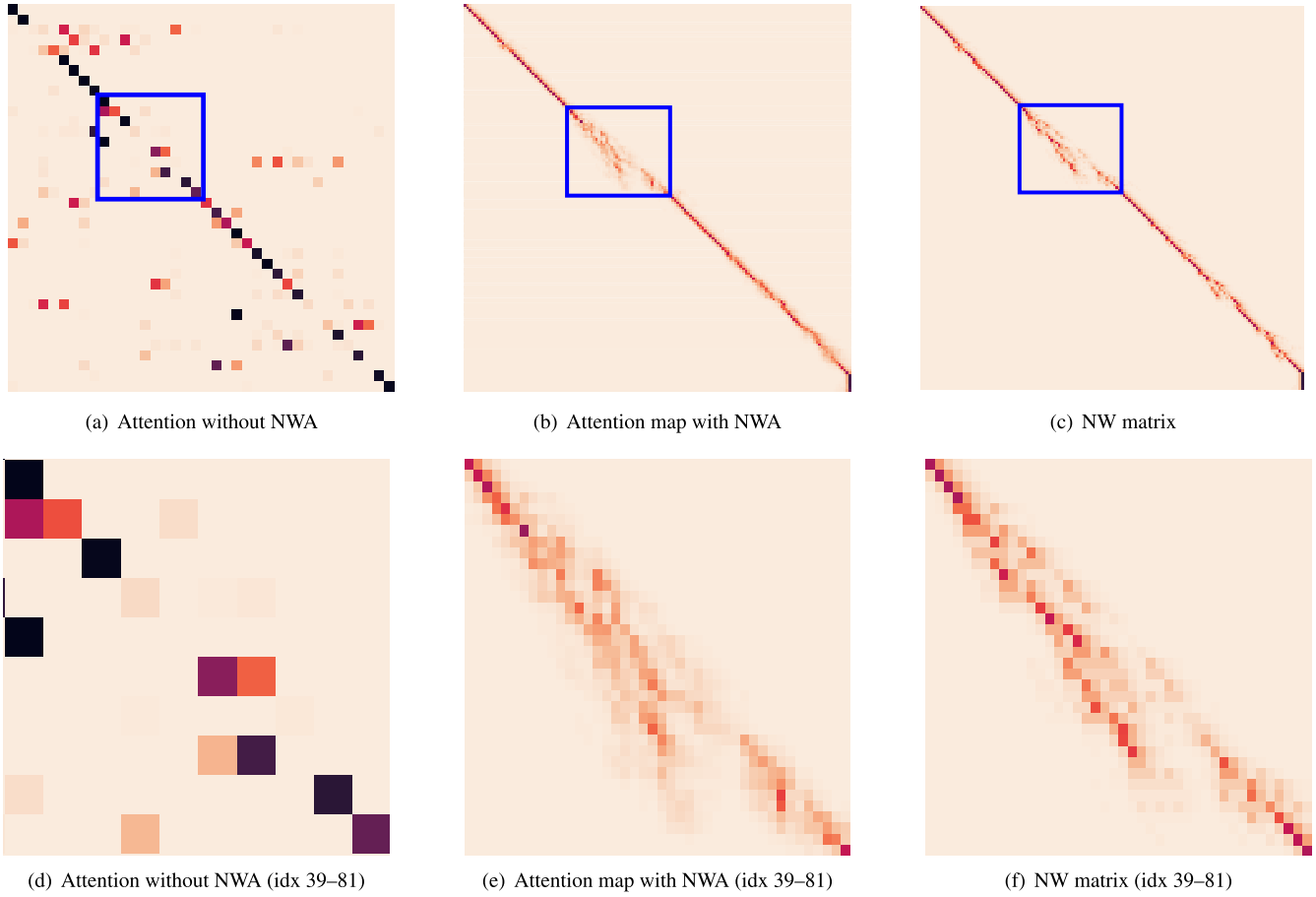
embedding dimension $m = 128$. Determining the optimal λ value involves consideration of both homologous and non-homologous pairs. The sweet spot of λ scales the exponent power of NW loss to $\frac{1}{10}$ of encoder loss. Consequently, we set the lambda value to 10^{-5}, a tenth of the converged encoder loss value (determined by MSE) for the Qiita dataset. Even though the selected λ value mostly captures the desired structures of edit distance and alignment, should this lambda scale factor of 10^{-5} prove unsuccessful, we opt for 10^{-6}, 10^{-8} respectively instead. For the DNA-Fountain dataset, setting the scale factor to 3 times λ in the Qiita setup turned out to be effective.

In terms of the details of the decoder and the cross-attention module, the decoder aligns with the Transformer encoder architecture [32], setting d_{model} to either 64 or 128, contingent upon the half-value of the embedding dimension m (i.e. $m/2$). The cross-attention module also adheres to this metric, assigning its hidden dimension as $m/2$. Both the decoder and the cross-attention module are equipped with 4 attention heads. Throughout all scenarios, the feedforward dimension of the decoder remains constant at 1024.

Due to resource constraints, we maintain a batch size of 1024. Parameter updates are executed using the RAdam optimizer [47]. In the initial 10 epochs, the learning rate is progressively amplified until it reaches its peak. From there, the learning rate decreases proportionally to the square root of the current epoch. The maximum learning rate is designated as 0.01 for encoder loss and 0.001 for $\mathcal{L}_{nw}$.

D. RESULTS

We conducted a comprehensive evaluation of various encoders, applying the Needleman-Wunsch Attention (NWA) framework in some instances, to demonstrate the improvements in embedding performance achieved through our



(a) Attention without NWA (b) Attention map with NWA (c) NW matrix

(d) Attention without NWA (idx 39–81) (e) Attention map with NWA (idx 39–81) (f) NW matrix (idx 39–81)

FIGURE 5. Visualization of the original Needleman-Wunsch (NW) matrix and the attention map trained with NWA.

TABLE 1. Root mean square error (RMSE) for qiita dataset.

Model	$m = 128$	$m = 128$ $(ED \leq K)$	$m = 256$	$m = 256$ $(ED \leq K)$
Transformer	1.84 ± 0.02	1.73 ± 0.05	1.75 ± 0.05	1.64 ± 0.02
Transformer + NWA	1.79 ± 0.01	1.63 ± 0.02	1.70 ± 0.02	1.62 ± 0.04
CNN	1.55 ± 0.01	1.29 ± 0.02	1.51 ± 0.01	1.32 ± 0.04
CNN + NWA	1.52 ± 0.01	1.29 ± 0.02	1.47 ± 0.01	1.27 ± 0.03
DSEE	3.92 ± 0.15	1.92 ± 0.05	3.99 ± 0.17	1.92 ± 0.06
DSEE + NWA	3.73 ± 0.05	1.85 ± 0.11	3.91 ± 0.15	1.89 ± 0.09

TABLE 2. RMSE for DNA-fountain dataset.

Model	$m = 128$	$m = 256$
Trans	2.01 ± 0.32	2.00 ± 0.33
Trans + NWA	1.74 ± 0.04	1.76 ± 0.06
CNN	1.61 ± 0.02	1.61 ± 0.01
CNN + NWA	1.61 ± 0.01	1.56 ± 0.04
DSEE	5.82 ± 0.07	6.51 ± 1.86
DSEE + NWA	5.54 ± 0.04	6.42 ± 1.29

proposed methodology. This evaluation included testing with and without NWA across different network architectures, focusing on embedding dimensions of $m = 128$ and $m = 256$. In this context, 'Transformer' denotes NeuroSEED [18] using a Transformer encoder, while 'CNN' refers to NeuroSEED with a CNN encoder. The reported Root Mean Square Error (RMSE) values are averages with their respective standard deviations, calculated over three trials. Detailed descriptions of the encoder networks used can be found in Sections V-B1 and V-B2. Additionally, we set the threshold K for identifying homologous pairs at 40, in accordance with the conventions established in DSEE.

Table 1 presents the root mean squared error (RMSE) and K-homologous RMSE for each model. The results indicate that the cross-attention module accurately aligns given sequence pairs, significantly enhancing the preservation of edit distance within the embedding. The embedding quality of Qiita DNA data showed improvement for both general pairs and homologous pairs. In instances where $m = 128, 256$, the NWA decoder reduced the RMSE by a maximum of 3% and 2%, respectively, compared to the encoder trained without NWA.

The evaluation results for the DNA-Fountain dataset are shown in Table 2. Notably, our framework demonstrates a remarkable improvement in the quality of sequence embeddings, achieving a maximum enhancement of 13% with the the Transformer encoder. Across all encoder structures employing the NWA, we observed consistent improvements in embedding quality, ranging from 3% to 4%. However, CNN architecture embedding with dimension $m = 128$ stood as an exception to this trend.

Table 2 does not include RMSE of homologous sequence pairs because the DNA-Fountain dataset used in our study involved random sampling of sequences. Consequently, the dataset comprises a limited number of homologous sequence pairs. Given this random sampling approach, Table 2 provides an approximation of the edit distance for the entire set of sequence pairs within the dataset. We hypothesize that the data distribution of the DNA-Fountain dataset is associated with a notable deviation and error in the DSEE setup.

VI. LEARNED ATTENTION MAP

As discussed in previous sections, the key idea of NWA is to utilize common features between the attention map and the normalized NW matrix, wherein similar portions of vectors or sequences tend to align. In this section, we show the effectiveness of the proposed scheme by presenting a visual comparison between the attention map and the NW matrix.

Fig. 5 provides a visual comparison between the original Needleman-Wunsch (NW) matrix and the attention map generated from the NWA for a pair of sequence samples. The attention map, as displayed in Fig. 5(b), evidently bears a close resemblance to the original NW matrix shown in Fig. 5(c). Notably, the NW matrix features a diffused region (highlighted by a blue box) due to the occurrence of numerous edit operations within subsequences ranging from the 39th to the 81st index of the original input sequences:

```
GGGG-A-AACGACATTGAGTGCTTGCACT-CTTTGG-GCGTCGAC-
GGGGCAGCACGA-ATT-A----GCAATAGTTTGGTG-G-CGACC
```

Fig. 5(e) and Fig. 5(f) zoom into the blue box area, presenting the learned attention map (NeuroSEED-CNN with NWA) and the NW matrix. The learned attention map successfully mirrors the target NW matrix, even within this noisy region.

For comparison, we also generated an attention map using the original NeuroSEED-Transformer without NWA, as shown in Fig. 5(a) and Fig. 5(d). As it does not include a dedicated decoder, we extracted the vectors preceding the encoder's linear projection. We then computed cross-attention by performing dot-product operations and applying softmax. It is crucial to emphasize that this encoder has never acquired information related to alignment structure. Given that the NeuroSEED-Transformer has a segment size of 4, the dimension of the attention map is $1/4 \times 1/4$ of the original map. It exhibits a diagonal structure, suggesting that the encoder also strives to match the corresponding components of sequences. However, it demonstrates a noisy pattern with large components for highly off-diagonal elements.

VII. CONCLUSION AND LIMITATION

Recognizing the computational challenges of sequence alignment and the limitations of existing deep learning-based edit distance-preserving embedding networks, we introduce the Needleman-Wunsch Attention (NWA) framework. This novel approach enables these networks to learn the alignment structure through NW attention, enhancing the embedding quality and accurately mapping the alignment distribution. Thus, NWA effectively preserves crucial alignment information in DNA sequences. Our observations of the alignment between the attention map and the NW matrix have led to the introduction of this framework. Moreover, NWA can be incorporated into any existing deep learning-based DNA sequence embedding network.

The NWA framework utilizes the attention mechanism to generate an attention map of given sequence pairs. Although the NWA successfully captures both edit distance structure and alignment distribution, NWA inherits the usage of the fixed sequence length of the attention. Such constraint limits thorough study between sequences with different or extremely long lengths. Future works should aim to devise a model that facilitates various lengths.

REFERENCES

[1] A. D. Baxevanis, G. D. Bader, and D. S. Wishart, *Bioinformatics*. Hoboken, NJ, USA: Wiley, 2020.

[2] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Commun. ACM*, vol. 7, no. 3, pp. 171–176, Mar. 1964.

[3] V. I. Levenshtein "Binary codes capable of correcting deletions, insertions, and reversals," *Sov. Phys. Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[4] A. J. Guo, C. Liang, and Q.-H. Hou, "Deep squared Euclidean approximation to the Levenshtein distance for dna storage," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 1–22.

[5] J. Jeong, S.-J. Park, J.-W. Kim, J.-S. No, H. H. Jeon, J. W. Lee, A. No, S. Kim, and H. Park, "Cooperative sequence clustering and decoding for DNA storage system with fountain codes," *Bioinformatics*, vol. 37, no. 19, pp. 3136–3143, Oct. 2021.

[6] C. Semple and M. Steel, *Phylogenetics*. London, U.K.: Oxford Univ. Press, 2003.

[7] J. Felsenstein, *Inferring Phylogenies*. Sunderland, MA, USA: Sinauer associates, 2004.

[8] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: An overview," *WIREs Data Mining Knowl. Discovery*, vol. 2, no. 1, pp. 86–97, Jan. 2012.

[9] K. Kukich, "Techniques for automatically correcting words in text," *ACM Comput. Surveys*, vol. 24, no. 4, pp. 377–439, Dec. 1992.

[10] G. Navarro, "A guided tour to approximate string matching," *ACM Comput. Surveys*, vol. 33, no. 1, pp. 31–88, Mar. 2001.

[11] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," *J. ACM*, vol. 21, no. 1, pp. 168–173, Jan. 1974.

[12] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *J. Mol. Biol.*, vol. 147, no. 1, pp. 195–197, Mar. 1981.

[13] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *J. Mol. Biol.*, vol. 48, no. 3, pp. 443–453, Mar. 1970.

[14] S. Kariin and C. Burge, "Dinucleotide relative abundance extremes: A genomic signature," *Trends Genet.*, vol. 11, no. 7, pp. 283–290, Jul. 1995.

[15] G. E. Sims, S.-R. Jun, G. A. Wu, and S.-H. Kim, "Alignment-free genome comparison with feature frequency profiles (FFP) and optimal resolutions," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 8, pp. 2677–2682, Feb. 2009.

[16] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. Inf. Theory*, vol. 37, no. 1, pp. 145–151, Jul. 1991.

[17] J. Chen, L. Yang, L. Li, S. Goodison, and Y. Sun, "Alignment-free comparison of metagenomics sequences via approximate string matching," *Bioinf. Adv.*, vol. 2, no. 1, Jan. 2022, Art. no. vbac077.

[18] G. Corso, Z. Ying, M. Pándy, P. Veličković, J. Leskovec, and P. Liò, "Neural distance embeddings for biological sequences," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 1–17.

[19] X. Dai, X. Yan, K. Zhou, Y. Wang, H. Yang, and J. Cheng, "Convolutional embedding for edit distance," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1–17.

[20] W. Zheng, L. Yang, R. J. Genco, J. Wactawski-Wende, M. Buck, and Y. Sun, "SENSE: Siamese neural network for sequence embedding and alignment-free comparison," *Bioinformatics*, vol. 35, no. 11, pp. 1820–1828, Jun. 2019.
[21] M. Nickel and D. Kiela, "Poincaré embeddings for learning hierarchical representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–11.
[22] A. Klimovskaia, D. Lopez-Paz, L. Bottou, and M. Nickel, "Poincaré maps for analyzing complex hierarchies in single-cell data," *Nature Commun.*, vol. 11, no. 1, p. 2966, Jun. 2020.
[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, vol. 1, 2019, pp. 1–15.
[24] T. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1–19.
[25] P.-H. Chi, P.-H. Chung, T.-H. Wu, C.-C. Hsieh, Y.-H. Chen, S.-W. Li, and H.-Y. Lee, "Audio albert: A lite BERT for self-supervised learning of audio representation," in *Proc. IEEE Spoken Lang. Technol. Workshop (SLT)*, Jan. 2021, pp. 1–12.
[26] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "LLaMA: Open and efficient foundation language models," 2023, *arXiv:2302.13971*.
[27] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi, "Self-instruct: Aligning language models with self-generated instructions," in *Proc. 61st Annu. Meeting Assoc. Comput. Linguistics*, 2023, pp. 1–14.
[28] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–58.
[29] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.
[30] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–17.
[31] M. Dehghani et al., "Scaling vision transformers to 22 billion parameters," in *Proc. 40th Int. Conf. Mach. Learn.*, 2023, pp. 1–16.
[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1–6.
[33] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1–18.
[34] H. Mi, Z. Wang, and A. Ittycheriah, "Supervised attentions for neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 15–17.
[35] S. Karlin and S. F. Altschul, "Applications and statistics for multiple high-scoring segments in molecular sequences," *Proc. Nat. Acad. Sci. USA*, vol. 90, no. 12, pp. 5873–5877, Jun. 1993.
[36] S. Koide, K. Kawano, and T. Kutsuna, "Neural edit operations for biological sequences," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–8.
[37] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.
[38] I. Chami, A. Gu, V. Chatziafratis, and C. Ré, "From trees to continuous embeddings and back: Hyperbolic hierarchical clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1–14.
[39] R. Chenna, "Multiple sequence alignment with the clustal series of programs," *Nucleic Acids Res.*, vol. 31, no. 13, pp. 3497–3500, Jul. 2003.
[40] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1–23.
[41] E. Englesson and H. Azizpour, "Generalized Jensen-Shannon divergence loss for learning with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 1–45.
[42] J. C. Clemente et al., "The microbiome of uncontacted amerindians," *Sci. Adv.*, vol. 1, no. 3, Apr. 2015, Art. no. e1500183.
[43] Y. Erlich and D. Zielinski, "DNA fountain enables a robust and efficient storage architecture," *Science*, vol. 355, no. 6328, pp. 950–954, Mar. 2017.
[44] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1–59.
[45] A. Haviv, O. Ram, O. Press, P. Izsak, and O. Levy, "Transformer language models without positional encodings still learn positional information," in *Findings of the Association for Computational Linguistics: EMNLP*. UAE, 2022, pp. 1382–1390.
[46] Solomon Kullback and Richard A Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.
[47] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–22.

KYELIM LEE received the B.S. degree in electronic and electrical engineering from Hongik University, Seoul, South Korea, in 2023, where he is currently pursuing the M.E. degree. His research interests include data embedding, language models, and bioinformatics.

ALBERT NO (Member, IEEE) received the B.S. degree in electrical engineering and mathematics from Seoul National University, Seoul, South Korea, in 2009, and the M.Sc. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 2012 and 2015, respectively. From 2015 to 2017, he was a Data Scientist with Roche. He was an Assistant Professor Hongik University, Seoul, in 2017. In 2024, he transitioned to Yonsei University, where he currently holds the position of an Associate Professor with the Department of Artificial Intelligence. His research interests include learning theory, differential privacy, lossy compression, and bioinformatics.