

Compare Similarities Between DNA Sequences Using Permutation-Invariant Quantum Kernel

Chenyu Shi¹, Gabriele Leoni², Mauro Petrillo³, Antonio Puertas Gallardo^{*2}, Hao Wang^{*1}

¹Leiden Institute of Advanced Computer Science, Leiden University, Leiden, the Netherlands

Email: {c.shi,h.wang}@liacs.leidenuniv.nl

²Joint Research Centre (JRC), European Commission, Ispra, Italy

Email: {gabriele.leoni,antonio.puertas-gallardo}@ec.europa.eu

³Seidor Italy S.r.l., Milan, Italy

Email: mauro.petrillo@seidor.com

Abstract—Computing the similarity between two DNA sequences is of vital importance in bioscience, yet it can be computationally expensive on classical hardware. For example, the edit distance with move operations (EDM), a DNA similarity measure of interest in biology, is proven to be NP-Complete to compute exactly on classical hardware. Recently, applied quantum algorithms have been anticipated to offer potential advantages over classical approaches. In this paper, we propose a novel variational quantum kernel model served as a surrogate model for estimating similarity between DNA sequences defined by EDM. Since the EDM metric exhibits a pairwise permutation-insensitive property, we incorporate a permutation-invariant structure into the variational quantum kernel to approximate this symmetry. Furthermore, to encode the four nucleotide bases as quantum states, we introduce a theoretically motivated encoding scheme based on symmetric informationally complete positive operator-valued measure (SIC-POVM) states. This encoding ensures mutual equivalence among bases, as each pair of symbols is mapped to quantum states that are equidistant on the Bloch sphere. We experimentally show that, equipped with the permutation-invariant circuit design and mutual-equivalence encoding, the proposed quantum kernel model achieves strong performance in approximating the similarity defined by EDM. Compared with classical kernel learning methods, our quantum approach achieves significantly higher accuracy while using substantially fewer trainable parameters.

Index Terms—Quantum Machine Learning; DNA Sequences Comparison; Quantum Kernel; Permutation Invariance;

I. INTRODUCTION

Comparing the similarity between DNA sequences is a fundamental task in bioinformatics and comparative genomics [1]. With the rapid development in bioscience techniques, the need for efficient and accurate DNA sequence similarity comparison has become increasingly important, particularly in applications such as antimicrobial resistance (AMR) gene detection [2], [3]. Numerous classical methods have been proposed for evaluating DNA sequence similarity. For example, the Needleman-Wunsch algorithm utilizes edit distance for global sequence alignment [4]. Additionally, FASTA [5] and BLAST [6] employ heuristic approaches to compute similarity, both of which have achieved significant success.

However, due to the large scale of DNA sequence data, many edit distance-based classical methods can be computationally

expensive and resource-consuming. For example, the edit distance with move operations (EDM), which is of great interest in biological sequence analysis, has been proven to be NP-Complete to compute exactly on classical hardware [7]. With the advancement of quantum computing, quantum computers are expected to have the potential to efficiently solve complex problems that are intractable for classical computers [8], [9]. Near-term quantum computing on the Noisy Intermediate-Scale Quantum (NISQ) devices [10] has already shown its ability in many fields, such as quantum chemistry [11], optimization [12], and quantum machine learning [13], [14]. In the task of DNA sequence comparison, quantum computing also holds promise for outperforming classical methods [15], [16].

We face two main challenges in designing the quantum model: (1) determining how to encode nucleotide bases onto a quantum computer in a principled manner; (2) preserving the pairwise permutation-insensitive property of the EDM metric within the model. Inspired from geometric machine learning [17], incorporating the intrinsic symmetries or invariances of data into a machine learning model is well known to enhance both performance and correctness. Examples include Convolutional Neural Networks (CNNs) [18], which leverage translation invariance in images; Graph Neural Networks (GNNs) [19], which leverage permutation invariance of graph nodes; and recent advances in quantum machine learning [20]–[22], where symmetries such as permutation invariance are similarly utilized.

In this work, we propose a novel permutation-invariant variational quantum kernel model specifically designed to learn the similarity between DNA sequences defined by EDM. We tackle the above two challenges and contribute as follows:

- For challenge (1), we develop a novel encoding circuit for four nucleotide bases to capture the mutual equality among them.
- For challenge (2), we consider DNA sequence similarity defined by EDM and design a permutation-invariant quantum kernel that approximately reflects the pairwise permutation insensitivity property of EDM.
- We conduct numerical experiments to validate the correctness and performance of our method on a classical

* Corresponding Authors

simulator and compare to classical deep kernel learning models.

- We incorporate the data re-uploading technique [23], [24] into our model and experimentally investigate how the expressiveness of the quantum kernel improves with increased data re-uploading.

Through theoretical analysis and experimental validation, our model has shown its effectiveness for DNA sequence comparison with polynomial circuit depth. Additionally, the broad applicability of kernel functions opens potential directions for future research and application in downstream tasks, such as improving sequence data analysis approaches for global challenges like antimicrobial resistance (AMR).

This paper is organized as follows: Section II overviews the fundamental background knowledge relevant to this work. Section III details the construction of our model, highlighting how it leverages symmetry in the encoding and parameterized circuits. Section IV presents and analyzes the results of numerical experiments. Section V discusses the model’s performance, limitations, and potential future directions.

II. PRELIMINARY

In this section, we provide the basic background information relevant to this work. Subsection II-A briefly introduces the concept of variational quantum computing, while Subsection II-B discusses the EDM, the metric used as the ground truth in this paper for measuring the similarity between DNA sequences. Subsection II-C emphasizes the pairwise permutation-insensitive symmetry of EDM.

A. Variational Quantum Computing

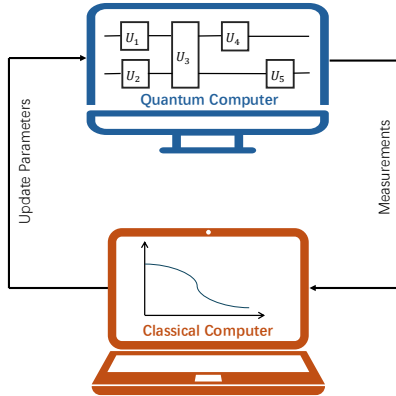


Fig. 1: The working paradigm of variational quantum computing. The quantum computer applies a variational quantum circuit to process information or construct models for a certain task. The measurement results will be fed to a classical computer for optimization. The optimizer on the classical computer will update the parameters in the quantum circuit. After multiple loops, the parameters in the quantum circuit will be adjusted to be suitable for solving the target task.

Variational quantum computing is a hybrid quantum-classical approach that leverages the strengths of both quantum

and classical computing. In this paradigm, a parameterized quantum circuit (also referred to as a variational quantum circuit) is employed to process information on quantum devices. The measurements from the quantum circuit are then passed to an optimizer on a classical computer for a specific target task, such as minimizing the expectation value of quantum measurements. The classical optimizer analyzes the quantum measurement output and updates the parameters of the quantum circuit accordingly. Through multiple iterations of this loop, the parameters are gradually adjusted to optimize performance for the target task.

Variational quantum computing has attracted significant research attention for the current NISQ devices [10]. Numerous algorithms based on variational quantum computing have been proposed, with the potential to achieve quantum advantage over classical methods. Notable examples include the Variational Quantum Eigensolver (VQE) [11] for quantum chemistry and the Quantum Approximate Optimization Algorithm (QAOA) [12] for optimization.

Among these algorithms, quantum machine learning has garnered much attention. In quantum machine learning, a quantum circuit consisting of a parameterized circuit $U(\theta)$ and an encoding circuit $V(x)$ is commonly used to construct a machine learning model f_θ . The expectation value of quantum measurements with respect to an observable O serves as the model’s output, which can be formulated as:

$$f_\theta(x) = \langle 0 | U^\dagger(\theta) V^\dagger(x) O V(x) U(\theta) | 0 \rangle \quad (1)$$

By feeding this output into a classical optimizer, the parameters of the model f_θ are iteratively updated to solve the target machine learning task. A number of variational quantum algorithms have been developed for machine learning applications, including the Variational Quantum Classifier [25], the Quantum Expectation Value Sampler [26], and the Variational Quantum Kernel [14]. In this work, we focus on the variational quantum kernel and introduce novel features specifically for DNA sequence comparison. The details of our method are discussed in Section III.

B. Similarity between DNA Sequences

Evaluating the similarity between DNA sequences is a fundamental task in bioinformatics and comparative genomics. The study of sequence comparison algorithms originated with the Needleman-Wunsch algorithm for global sequence alignment [4], which is a generalization of the Levenshtein edit distance [27].

The Levenshtein edit distance between two strings is defined as the minimum number of edits required to transform one string into the other, where the allowed operations include insertion, deletion, and substitution of a single character. Under certain parameter settings, the Needleman-Wunsch algorithm is equivalent to computing the Levenshtein distance between two DNA sequences [28]. Therefore, Levenshtein distance can be applied as a metric to measure the similarity between DNA sequences.

Based on the standard Levenshtein distance, the edit distance with move operations (EDM) has been of interest in biological sequence analysis [7]. In addition to the character-wise operations allowed in standard Levenshtein distance, EDM further permits substring-move operations. However, the exact classical computation of EDM has been proven to be NP-Complete [7], which poses a significant obstacle to applying EDM for measuring the similarity between DNA sequences. In this work, we use this computationally challenging similarity measure as the ground truth for evaluating the similarity between pairs of DNA sequences.

C. Permutation Insensitivity in EDM

In this subsection, we emphasize the pairwise permutation insensitivity inherent in EDM. Let $x = [x_1, x_2, \dots, x_n]$ and $y = [y_1, y_2, \dots, y_n]$ be two DNA sequences of length n , and $D_E(x, y)$ be EDM between x and y . The pairwise permutation operator Π_{ij} ($1 \leq i, j \leq n$) represents the operation of swapping the bases at positions i and j . For example, applying Π_{12} to x results in $\Pi_{12}(x) = [x_2, x_1, \dots, x_n]$. It can be verified that EDM changes by at most 2 when **the same** pairwise permutation is applied to both input sequences x and y :

$$|D_E(x, y) - D_E(\Pi_{ij}(x), \Pi_{ij}(y))| \leq 2 \quad (2)$$

This kind of property is known as pairwise permutation insensitivity [29], [30]. We provide a simple proof of this conclusion in Equation 2. Let $L = D_E(x, y)$ while $L' = D_E(\Pi_{ij}(x), \Pi_{ij}(y))$. Note that transforming x into $\Pi_{ij}(x)$ requires only one move operations at most, and the same holds for transforming $\Pi_{ij}(x)$ to x . The same reasoning also applies to y . If $L > L'$, then we can always transform x and y into $\Pi_{ij}(x)$ and $\Pi_{ij}(y)$ by at most 2 move operations in total, which indicates that $L - L' \leq 2$. Similarly, if $L < L'$, we can derive that $L' - L \leq 2$. Therefore, we conclude that Equation 2 holds. Details in Subsection III-C, our model is specifically designed to approximately capture the pairwise permutation-insensitive property of EDM, aiming to achieve potential performance advantages.

III. METHODOLOGY

In this section, we use the most abstract variational quantum kernel model as the starting point and show how to incrementally construct our model from sketch to the final concrete version. The symmetric characteristics in the DNA data comparison task taken advantage of in our model will be discussed in detail.

A. Method Sketch: Variational Quantum Kernel

Our model sketch begins with the most abstract representation of a variational quantum kernel, as illustrated in Figure 2. The variational quantum kernel is a specialized form of a variational quantum circuit, consisting of a parameterized layer $U(\theta)$, an encoding layer $V(\cdot)$, and their corresponding conjugate transpose layers $V^\dagger(\cdot)$ and $U^\dagger(\theta)$. The initial state is the zero state $|0\rangle$. The probability of measuring 0 is used as the output of the quantum model.

For a pair of inputs x and y , the output of the quantum model K_θ is given by:

$$\begin{aligned} K_\theta(x, y) &= |\langle 0| U^\dagger(\theta) V^\dagger(y) V(x) U(\theta) |0\rangle|^2 \\ &= \text{tr}(\rho_\theta(x) \rho_\theta(y)) \end{aligned} \quad (3)$$

where the first equality follows from the Born rule [31], while the third equality uses the density operator representation of quantum states. Specifically, $\rho_\theta(x) = V(x)U(\theta)|0\rangle\langle 0|U^\dagger(\theta)V^\dagger(x)$ and $\rho_\theta(y) = V(y)U(\theta)|0\rangle\langle 0|U^\dagger(\theta)V^\dagger(y)$.

Note that $\rho_\theta(\cdot)$ can be viewed as a parameterized feature mapping function from the input space $\mathcal{X}$ to the feature space. According to the definition of the Frobenius inner product, the output of the model $K_\theta(x, y)$ represents the inner product between the density matrices $\rho_\theta(x)$ and $\rho_\theta(y)$. Therefore, by the definition of the kernel method [32], the quantum model $K_\theta : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ qualifies as a kernel function.

In our case, the inputs x and y represent two DNA sequences to be compared. The quantum kernel K_θ evaluates the similarity between x and y and outputs a score $K_\theta(x, y)$ between 0 and 1. The more similar the two sequences are, the higher the score will be. Notably, if the two input DNA sequences are identical, the output score will be 1.

In the following subsections, we will construct this abstract model step by step, methodically building from the foundation to the final concrete version tailored to our task.

B. Encoding Layer: Mutual Equality

In this subsection, we will construct the encoding layer $V(\cdot)$ for our task. A DNA sequence can be represented as a string, where each position corresponds to one of the four nucleotide bases: A, T, G, or C. Our overall strategy of encoding is to encode each base using a single qubit and stack these qubits together to represent the entire DNA sequence. For each base, we map it to a unique single-qubit state. The specific quantum states associated with each base are shown in Table I. These states are known as the SIC-POVM states for a single-qubit quantum system [33].

Nucleotide Bases	Quantum state representation
Adenine (A)	$ 0\rangle$
Thymine (T)	$\frac{1}{\sqrt{3}} 0\rangle + \sqrt{\frac{2}{3}} 1\rangle$
Guanine (G)	$\frac{1}{\sqrt{3}} 0\rangle + \sqrt{\frac{2}{3}}e^{i\frac{2\pi}{3}} 1\rangle$
Cytosine (C)	$\frac{1}{\sqrt{3}} 0\rangle + \sqrt{\frac{2}{3}}e^{i\frac{4\pi}{3}} 1\rangle$

TABLE I: In the encoding layer, each nucleotide base is assigned a unique quantum state (SIC-POVM).

An important feature of these states is their symmetric property, which allows us to capture the mutual equality when encoding the four bases. Simply put, without biases of the parameterized layers, the similarity between A and G should not be greater than the similarity between A and C. Therefore, in each qubit, the encoding layer must treat the four bases equally, ensuring that in the encoding space, the distance

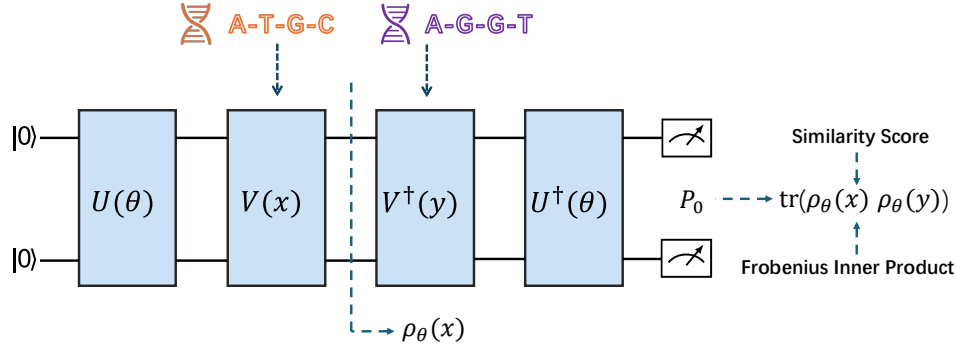


Fig. 2: The model sketch of the variational quantum kernel. The model sketch of the variational quantum kernel consists of a parameterized layer $U(\theta)$, an encoding layer $V(\cdot)$, and their corresponding conjugate transpose layers, arranged sequentially. The initial state is set to be $|0\rangle$. The output of the model is the probability of measuring 0 in the computational basis, which is the Frobenius inner product between two density matrices. Thus, the entire model constructs a kernel function to metric similarity using a variational quantum circuit.

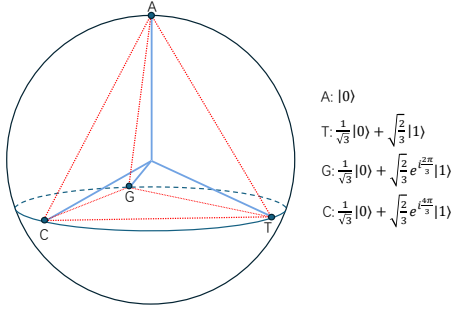


Fig. 3: The four SIC-POVM states form a regular tetrahedron on the Bloch sphere.

between any pair of the quantum states representing the four bases is identical.

It is evident that the SIC-POVM states fulfill this requirement. As shown in Figure 3, the four states form a regular tetrahedron on the Bloch sphere, with each vertex representing a state (corresponding to a base). This highly symmetric property ensures that our encoding layer maintains the mutual equality among the four bases on each qubit.

This method also takes advantage of the biological characteristics of DNA sequences. Notably, there can be at most four states on the Bloch sphere that satisfy the condition of mutual equality. This can be understood geometrically: in three-dimensional space, there are at most four points where the distance between any two points is equal. For DNA sequences, there are exactly four bases, which makes our strategy of encoding effective. In Figure 4, we provide an illustrative example of a quantum circuit used to encode the length-4 DNA sequence “ATGC” following our encoding strategy.

C. Parameterized Layer: Permutation Invariance

In this subsection, we construct the encoding layer $U(\theta)$ for our task. As mentioned in Section II-B, the similarities

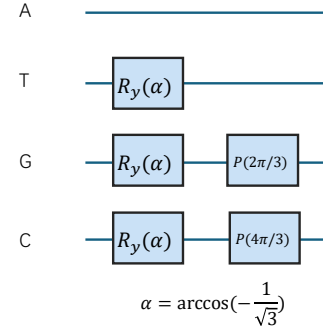


Fig. 4: The encoding layer for the length-4 DNA sequence “ATGC”. For Adenine, there is no quantum gate because only an identity operator is needed. For the other bases, a rotation gate R_y and a phase gate P are applied with corresponding angles.

between two DNA sequences can be quantified using the edit distance. In our case, we primarily consider EDM D_E as the metric.

As discussed in Section II-C, EDM exhibits a pairwise permutation-insensitive symmetry. A direct implication is that a well-designed kernel for evaluating the similarity between two DNA sequences should exhibit the same symmetry. However, due to the difficulty of directly incorporating permutation insensitivity into the model, we use permutation invariance to approximately capture the desired permutation-insensitive symmetry. Therefore, the variational quantum kernel K_θ used in our task is expected to satisfy the pairwise permutation-invariant property:

$$K_\theta(x, y) = K_\theta(\Pi_{ij}(x), \Pi_{ij}(y)) \quad (4)$$

Permutation invariance is not a universal property of variational quantum circuits or quantum kernels. Therefore, to ensure that the quantum kernel exhibits this property, we must carefully design the quantum circuit.

Since there are no entanglement gates in the encoding layer, we can verify that if the parameterized circuit U_θ is permutation-invariant, then the quantum kernel will satisfy Equation (4). Following the definition in [22], a quantum circuit U is said to be permutation-invariant if it satisfies the condition:

$$U = U_{\Pi_{ij}} U U_{\Pi_{ij}} \quad (5)$$

where $U_{\Pi_{ij}}$ represents the SWAP gate acting on the i -th and j -th qubits. Furthermore, since there are no entanglement gates in the encoding layer, it is straightforward to verify that:

$$V(\Pi_{ij}(x)) = U_{\Pi_{ij}} V(x) U_{\Pi_{ij}} \quad (6)$$

Suppose the parameterized layer is permutation-invariant, following the Equation (5) and Equation (6), one can verify that the quantum kernel fulfills eq. (4), which ensures permutation invariance.

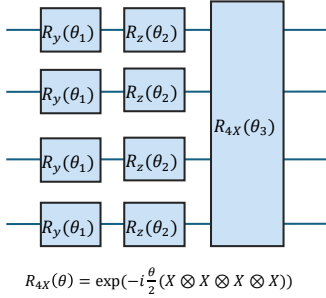


Fig. 5: The permutation-invariant parameterized layer for the length-4 DNA sequence. The R_{NX} gate is applied to provide entanglement, after a series of R_z and R_y gates.

Following the design principles for permutation-invariant circuits outlined in [22], we use the following simple quantum circuit as the parameterized layer. A 4-qubit example is illustrated in Figure 5. The circuit consists of an R_{NX} gate to introduce entanglement. The R_{NX} gate can be represented as $R_{NX}(\theta) = \exp(-i \frac{\theta}{2} (X \otimes \dots \otimes X))$. Then, a single qubit rotation gate R_z and R_y are applied. Specifically, the parameterized layer provides three trainable parameters. One can verify that this circuit satisfies the permutation-invariant property defined by Equation (4).

D. Data Re-uploading: Expressiveness

Instead of using only a single parameterized layer and one encoding layer with their conjugate transposes, we can incorporate multiple layers to enhance the model's expressive capability, following the data re-uploading technique [23], [24]. The data re-uploading technique alternates between different parameterized layers and the same encoding layer, which has been proven to improve the model's expressiveness. Besides, one can verify that applying the data re-uploading technique does not break the permutation-invariant property of the quantum kernel.

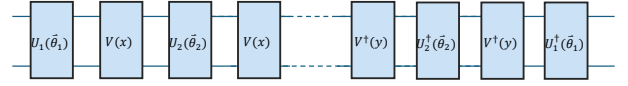


Fig. 6: The data re-uploading technique alternates between parameterized layers and encoding layers. It has been proven to improve the model's expressive capability. Notably, the conjugate transpose components also appear alternately in the correct order, ensuring that the permutation-invariant property is preserved.

E. The Model: Put It All Together

By incorporating all the specific details discussed above into the abstract sketch from Section III-A, we obtain our final model. First, the model employs the encoding layer from Section III-B to capture the mutual equality among nucleotide bases. Next, it utilizes a permutation-invariant quantum circuit as the parameterized layer. It ensures that the entire quantum kernel maintains permutation invariance, which approximately captures the essential permutation insensitivity property for measuring the similarity between DNA sequences. Finally, the data re-uploading technique is applied to enhance the model's expressiveness.

On a NISQ device, quantum circuits are generally limited to shallow depths that grow no faster than polynomially with the system size n . In our model, n corresponds to the length of the DNA sequences. Therefore, the computational complexity of evaluating our quantum kernel—i.e., computing a single approximate similarity score between two DNA sequences—remains polynomial to the sequence length n .

IV. NUMERICAL EXPERIMENTS

In this section, we evaluate the performance of the permutation-invariant quantum kernel in comparing the similarity between DNA sequences by numerical experiments on classical simulators. The experimental setup is detailed in Section IV-A, and the results are presented in Section IV-C. We also provide a brief introduction to the classical kernel learning models used for comparison in Section IV-B.

A. Experiment Setting

Due to the limitations of current quantum devices and classical simulators, our numerical experiments are restricted to comparing short DNA sequences, each consisting of eight nucleotide bases. All experimental results are obtained through classical simulations.

As discussed in Section III-E, since the DNA sequences have a length of 8, the quantum kernel in our experiments utilizes eight qubits. To enhance the model's expressive capability, the number of data re-uploading layers is set to 24, resulting in 72 trainable parameters in the quantum kernel model. In order to evaluate the effect of data re-uploading technology for improving the model's expressiveness, we also chose the number of data re-uploading as 6 and 12 to run the experiments for a comparison to 24. In the following, we

denote our method as QKernel- N , where N represents the number of data re-uploading layers.

For many tasks in bioscience, relative similarity scores are sufficient in practical applications. For example, when searching for the most similar sequences in a gene sequence repository for a given query DNA sequence A , it is only necessary to identify which sequence has the highest similarity with A . In this case, the absolute value of the similarity score is less important, whereas the relative ranking of similarity scores is crucial. Inspired by this fact, we propose using **order accuracy** as the metric to evaluate the model's performance.

Order accuracy evaluates whether the model preserves the similarity score rankings between two pairs of sequences. Specifically, for three DNA sequences A , B and C , if the ground truth similarity score (EDM D_E in this paper) shows the relation $D_E(A, B) > D_E(A, C)$, then a well-performing model should maintain the relative order, meaning that $K_\theta(A, B) < K_\theta(A, C)$. Note that similarity scores and the distances are inversely related. Order accuracy quantifies the percentage of DNA sequence triplets for which the quantum kernel model correctly preserves the relative similarity rankings.

The training and test set each contain 3200 triplets of DNA sequences, totaling 9600 sequences per set. The ground truth for training is based on EDM, which is transformed and normalized into a similarity score between 0 and 1 as follows:

$$S_T(A, B) = \frac{N - D_E(A, B)}{N} \quad (7)$$

where N is the length of DNA sequences (namely 8 in our experiments), and $D_E(A, B)$ denotes EDM between DNA sequences A and B . S_T serves as the ground truth similarity score in our experiments.

The optimizer used for training is the commonly used stochastic gradient descent (SGD) [34] with a learning rate of 0.01. The loss function is the mean square error (MSE) between the ground truth and the model's prediction. The model is trained for 100 epochs, and its performance is evaluated on the test set after each epoch.

B. Classical Deep Kernel Learning Models for Comparison

The quantum kernel model we propose can be regarded as a type of kernel learning model [35]. To assess the efficiency of our approach, we compare it with classical deep kernel learning models. To ensure a fair comparison to the greatest extent possible, we restrict the classical models to simple architectures (as illustrated in Figure 7), even though they already contain far more trainable parameters than our quantum kernel model (as reported in Table II).

The classical kernel learning model used for comparison consists of a parameterized feature map that embeds each DNA sequence into a vector representation. A kernel function is then applied to these embeddings to produce a similarity score. In our experiments, we consider three kernel functions: RBF, cosine, and poly2.

The structure of the parameterized feature map used in the comparison experiments is shown in Table II. First, the

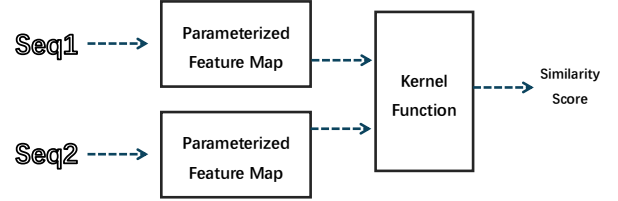


Fig. 7: The classical deep kernel learning models used for comparison. Each pair of sequences is embedded through a parameterized feature map, which is detailed in Table II. The resulting embeddings are then fed into a kernel function to produce the final similarity score. For the kernel functions, we use RBF, cosine and poly2 for comparison.

trainable embedding layer maps each of the four nucleotides to a 4-dimensional vector. Hence, a length-8 DNA sequence is represented as an $(8, 4)$ -tensor after embedding. This tensor is then flattened into a 32-dimensional vector, which serves as the input to a two-layer MLP. Although this parameterized feature map has a simple architecture, it already contains far more trainable parameters than our quantum kernel model.

Layers	Output Size	Number of Trainable Parameters
Embedding	$(8, 4)$	16
Flatten	$(32,)$	0
Linear	$(16,)$	528
ReLU	$(16,)$	0
Linear	$(16,)$	272

TABLE II: The parameterized feature map used in the classical deep kernel learning models shown in Figure 7. The settings are for length-8 DNA sequences, which contain 816 trainable parameters in total.

C. Results

The training and evaluation process is conducted independently 10 times to mitigate the impact of randomness. For QKernel-24, the average best-so-far learning curve along with the 95% confidence interval on the test set, is shown in Figure 8. We observe that the average order accuracy increases throughout the training process. The model improves rapidly at the beginning of training, followed by a gradual improvement at a slower rate.

The Qkernel-24 model begins with random guessing. For order accuracy, a completely random guess would result in 50% accuracy. After 100 epochs of training, the model achieves an average order accuracy of over 75%, which indicates that it successfully learns the characteristics of similarity between DNA sequences during training.

For quantum kernel models with different numbers of data re-uploading and for the classical deep kernel learning models, we use the same training and evaluation settings. The averaged best order accuracies achieved over 10 runs, together with

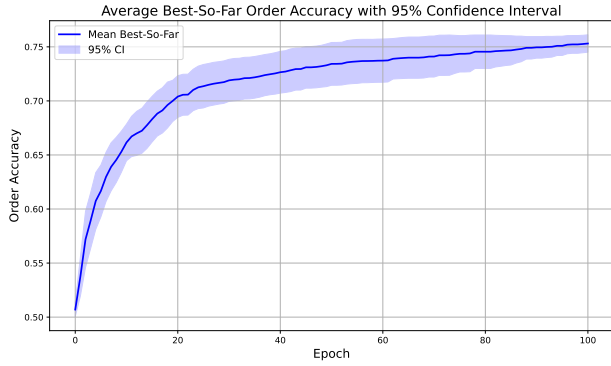


Fig. 8: Best-so-far learning curves with confidence interval. As the metric, the order accuracy is evaluated at the end of each training epoch. The model learns quickly at the beginning and improves at a slower rate in the later epochs. After 100 epochs of training, the model achieves an order accuracy of over 75%, indicating that it effectively learns during training.

the total number of trainable parameters in each model, are reported in Table III.

Kernel Learning Model	Order Accuracy	Number of Parameters
QKernel-24	$75.3 \pm 1.3\%$	72
Qkernel-12	$73.8 \pm 3.2\%$	36
Qkernel-6	$66.7 \pm 2.9\%$	18
Ckernel-RBF	$59.1 \pm 1.3\%$	817
Ckernel-cosine	$58.9 \pm 1.1\%$	816
Ckernel-poly2	$59.7 \pm 0.77\%$	818

TABLE III: The average best order accuracy achieved over 10 runs, corresponds to different data re-uploading numbers and different classical kernel learning models. We observe that order accuracy increases as the number of data re-uploading increases, indicating that the data re-uploading technique effectively enhances the expressiveness of the quantum kernel model. Besides, although classical kernel learning model contains more trainable parameters, its performance is far worse compared to the quantum kernel.

For quantum kernels, we observe that the order accuracy increases as the number of data re-uploading layers grows, indicating that the data re-uploading technique effectively enhances the expressive capacity of the model. Therefore, our quantum kernel is expected to achieve better performance with more data re-uploading layers.

For classical deep kernel learning models, we find that they perform significantly worse than the quantum kernel, even though they contain far more trainable parameters. The averaged order accuracies of all classical models remain around 60%. These results indicate that the specially designed permutation-invariant quantum kernel with mutually equivalent encoding substantially improves the efficiency of kernel learning. With far fewer parameters, our quantum kernel achieves considerably more promising results.

V. DISCUSSION AND CONCLUSION

In this work, we construct a permutation-invariant quantum kernel designed to compare the similarity between DNA sequences while leveraging the intrinsic pairwise permutation-insensitive symmetry of EDM. We develop an encoding circuit tailored for nucleotide bases, ensuring that it captures their mutual equality. To realize permutation invariance in the quantum kernel, we incorporate a permutation-invariant circuit. Additionally, we employ the data re-uploading technique to enhance the model’s expressivity. Numerical experiments indicate that despite utilizing only a few trainable parameters, the quantum kernel effectively measures the relative similarity between DNA sequences.

EDM is NP-Complete to compute exactly in the general case. Our quantum kernel model can be viewed as a surrogate model that approximates the similarity measure defined by EDM while allowing polynomial-time inference. It is worth noting that this comparison is not entirely fair, since the quantum kernel model requires a training phase before inference. Therefore, we additionally compare our method with several classical deep kernel learning models under the same training budget. Our quantum kernel consistently achieves superior order accuracy, even though the classical models employ far more trainable parameters. This superior performance likely stems from the fact that our quantum kernel explicitly captures permutation insensitivity in the data through its permutation-invariant structure, whereas the baseline classical neural networks must learn this symmetry from data.

Our model still has certain limitations due to the constraints of current quantum devices. The scalability of quantum devices is hindered by factors such as noise [10] and the barren plateaus problem [36], making it currently almost impossible to process real DNA sequences directly. This is primarily due to the circuit width required to represent the sequences. In our work, we only evaluate the model on DNA sequences of length 8. Although the model should theoretically remain effective for longer sequences, its performance has yet to be experimentally verified on large-scale data.

The scalability limitations of NISQ devices also constrain the circuit depth. In our work, we employ a relatively simple permutation-invariant parameterized circuit along with a limited number of data re-uploading iterations. While this simplification improves practicality on current hardware, it inevitably limits the model’s expressive capacity. As a result, the absolute order accuracy achieved by our method is not yet high. However, it still surpasses the order accuracy achieved by the baseline classical kernel learning models with larger parameter counts. Our experiments further indicate that increasing the number of data re-uploading layers enhances the performance of the model. Also, our method is expected to benefit from more sophisticated permutation-invariant circuit architectures for higher expressivity. General principles for constructing such expressive permutation-invariant quantum circuits can be found in [22].

In this work, we use permutation invariance in the design of

the quantum circuit to approximately capture the permutation insensitivity inherent in EDM between DNA sequences. However, this approximation may introduce error, as the desired symmetry is not exactly preserved. Designing a permutation-insensitive quantum kernel model directly instead of permutation invariance is more complex, but remains a meaningful direction for future work.

Like other general kernel methods, our quantum kernel model is theoretically applicable to a variety of downstream tasks in bioscience. The most direct application is identifying genes of interest (i.e., those similar to a reference DNA sequence) in gene sequence data repositories, including antimicrobial resistance (AMR) gene detection [2], [3], novel gene identification [37], and homology search [38]. Additionally, as a kernel function, our model naturally integrates with support vector machines (SVMs) to build classification models for bioscience applications, such as AMR gene classification [2]. Validation of the model on downstream tasks is beyond the scope of this paper. However, we consider these applications as potential directions for future work.

REFERENCES

- [1] J. M. Heather and B. Chain, "The sequence of sequencers: The history of sequencing dna," *Genomics*, vol. 107, no. 1, pp. 1–8, 2016.
- [2] M. Feldgarden, V. Brover, N. Gonzalez-Escalona, J. G. Frye, J. Haendiges, D. H. Haft, M. Hoffmann, J. B. Pettengill, A. B. Prasad, G. E. Tillman *et al.*, "Amrfinderplus and the reference gene catalog facilitate examination of the genomic links among antimicrobial resistance, stress response, and virulence," *Scientific reports*, vol. 11, no. 1, p. 12728, 2021.
- [3] B. S. Galhano, R. G. Ferrari, P. Panzenhagen, A. C. S. de Jesus, and C. A. Conte-Junior, "Antimicrobial resistance gene detection methods for bacteria in animal-based foods: A brief review of highlights and advantages," *Microorganisms*, vol. 9, no. 5, p. 923, 2021.
- [4] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of molecular biology*, vol. 48, no. 3, pp. 443–453, 1970.
- [5] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison," *Proceedings of the National Academy of Sciences*, vol. 85, no. 8, pp. 2444–2448, 1988.
- [6] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of molecular biology*, vol. 215, no. 3, pp. 403–410, 1990.
- [7] D. Shapira and J. A. Storer, "Edit distance with move operations," in *Annual Symposium on Combinatorial Pattern Matching*. Springer, 2002, pp. 85–98.
- [8] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, 1996, pp. 212–219.
- [9] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th annual symposium on foundations of computer science*. Ieee, 1994, pp. 124–134.
- [10] J. Preskill, "Quantum computing in the nisy era and beyond," *Quantum*, vol. 2, p. 79, 2018.
- [11] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature communications*, vol. 5, no. 1, p. 4213, 2014.
- [12] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.
- [13] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.
- [14] S. Jerbi, L. J. Fiderer, H. Poulsen Nautrup, J. M. Kübler, H. J. Briegel, and V. Dunjko, "Quantum machine learning beyond kernel methods," *Nature Communications*, vol. 14, no. 1, pp. 1–8, 2023.
- [15] B. Kösoğlu-Kind, R. Loredó, M. Grossi, C. Bernecker, J. M. Burks, and R. Buchkremer, "A biological sequence comparison algorithm using quantum computers," *Scientific Reports*, vol. 13, no. 1, p. 14552, 2023.
- [16] G. D. Varsamis, I. G. Karafyllidis, K. Gilkes, U. Arranz, R. Martín-Cuevas, G. Calleja, P. Dimitrakis, P. Kolovos, R. Sandaltzopoulos, H. Jessen *et al.*, "Quantum gate algorithm for reference-guided dna sequence alignment," *Computational Biology and Chemistry*, vol. 107, p. 107959, 2023.
- [17] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," *arXiv preprint arXiv:2104.13478*, 2021.
- [18] K. O'Shea, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.
- [19] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [20] P. Kairon, J. Jäger, and R. V. Krems, "Equivalence between exponential concentration in quantum machine learning kernels and barren plateaus in variational algorithms," *arXiv preprint arXiv:2501.07433*, 2025.
- [21] L. Schatzki, M. Larocca, Q. T. Nguyen, F. Sauvage, and M. Cerezo, "Theoretical guarantees for permutation-equivariant quantum neural networks," *npj Quantum Information*, vol. 10, no. 1, p. 12, 2024.
- [22] M. Mansky, S. L. Castillo, C. Linnhoff-Popien, and V. R. Puigvert, "Permutation-invariant quantum circuits," *Bulletin of the American Physical Society*, 2024.
- [23] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, "Data re-uploading for a universal quantum classifier," *Quantum*, vol. 4, p. 226, 2020.
- [24] A. Pérez-Salinas, D. López-Núñez, A. García-Sáez, P. Forn-Díaz, and J. I. Latorre, "One qubit as a universal approximant," *Physical Review A*, vol. 104, no. 1, p. 012405, 2021.
- [25] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," *arXiv preprint arXiv:1802.06002*, 2018.
- [26] J. Romero and A. Aspuru-Guzik, "Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions," *Advanced Quantum Technologies*, vol. 4, no. 1, p. 2000003, 2021.
- [27] V. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Proceedings of the Soviet physics doklady*, 1966.
- [28] P. H. Sellers, "On the theory and computation of evolutionary distances," *SIAM Journal on Applied Mathematics*, vol. 26, no. 4, pp. 787–793, 1974.
- [29] I. Roy, A. De, and S. Chakrabarti, "Adversarial permutation guided node representations for link prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 11, 2021, pp. 9445–9453.
- [30] R. Grainger, T. Paniagua, X. Song, N. Cuntoor, M. W. Lee, and T. Wu, "Paca-vit: learning patch-to-cluster attention in vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 18 568–18 578.
- [31] M. A. Nielsen and I. L. Chuang, *Quantum computation and quantum information*. Cambridge university press Cambridge, 2001, vol. 2.
- [32] J. Shawe-Taylor, "Kernel methods for pattern analysis," *Cambridge University Press google schola*, vol. 2, pp. 181–201, 2004.
- [33] J. M. Renes, R. Blume-Kohout, A. J. Scott, and C. M. Caves, "Symmetric informationally complete quantum measurements," *Journal of Mathematical Physics*, vol. 45, no. 6, pp. 2171–2180, 2004.
- [34] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.
- [35] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing, "Deep kernel learning," in *Artificial intelligence and statistics*. PMLR, 2016, pp. 370–378.
- [36] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature communications*, vol. 9, no. 1, p. 4812, 2018.
- [37] S. Klasberg, T. Bitard-Feildel, and L. Mallet, "Computational identification of novel genes: current and future perspectives," *Bioinformatics and Biology insights*, vol. 10, pp. BBI-S39950, 2016.
- [38] G. R. Reeck, C. de Haën, D. C. Teller, R. F. Doolittle, W. M. Fitch, R. E. Dickerson, P. Chambon, A. D. McLachlan, E. Margoliash, T. H. Jukes *et al.*, "'homology' in proteins and nucleic acids: a terminology muddle and a way out of it," *Cell*, vol. 50, no. 5, p. 667, 1987.