International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 11 | Nov - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Quantum AI Framework for Protein Structure Folding Prediction Using Quantum Neural Networks

N. Vijay Kiran[1]
UG Student
Dept. of CSE (AI & ML)
CMR Technical Campus
Hyderabad, Telangana, India
nvijaykiran04@gmail.com

Md. Adnan[2]
UG Student
Dept. of CSE (AI & ML)
CMR Technical Campus
Hyderabad, Telangana, India
mohdadnanmohi@gmail.com

Ch. Manikanta[3]
UG Student
Dept. of CSE (AI & ML)
CMR Technical Campus
Hyderabad, Telangana, India
chiluverimanikanta19@gmail.com

***Abstract*—Protein structure prediction is an important problem in bioinformatics, because the structure of a protein largely determines its biological function. Experimental methods give accurate structures but are expensive and slow, and large deep learning models need heavy computation. This paper presents a hybrid quantum-classical framework that predicts protein secondary structure, a step toward full folding prediction, from amino acid sequences. Sequences from the CB513 benchmark are validated, encoded numerically and reduced to a compact feature vector with principal component analysis (PCA). A Hybrid Quantum Neural Network (HQNN) then processes this vector: a classical encoder transforms the features, a variational quantum circuit with trainable rotation and entangling gates produces new features through measurement, and a classical layer assigns each residue to Helix (H), Sheet (E) or Coil (C). The quantum circuit runs on a simulator, so no quantum hardware is needed. On CB513 the model reaches 87.6% training accuracy and 82.4% validation/test accuracy. A web application takes a sequence as input and returns the predicted structure labels. The results show that a small simulated HQNN can learn the three secondary structure classes; they are not evidence of quantum advantage.**

***Index Terms*—protein structure prediction, secondary structure, CB513, quantum neural network, quantum machine learning, hybrid quantum-classical model, PCA.**

I. INTRODUCTION

Proteins are essential molecules in the body. They act as enzymes, carry cell signals, support the immune response and repair tissue. What a protein does depends on how its amino acid sequence folds into a three-dimensional structure, and incorrect folding is linked to serious diseases such as cancer. Protein structure is usually described at four levels: primary (the amino acid sequence), secondary, tertiary and quaternary. Secondary structure describes the local arrangement of the chain and is made up of alpha-helices, beta-sheets and coil regions [1].

Predicting structure from the amino acid sequence alone is a major problem in bioinformatics and computational biology. A reliable prediction helps in understanding what proteins do, how they are shaped, how drugs can be designed, and how the structures of related proteins can be inferred. The problem is hard because the primary sequence does not show the structure directly. The structural state of a residue depends on the residue itself, on its neighbours and on the pattern of the whole sequence. Deciding whether each residue belongs to a Helix, a Sheet or a Coil is like solving a puzzle, but an accurate answer gives useful information before any more complicated analysis.

Many computational methods have been proposed. Early methods used the statistical properties of amino acids [2]. Later work applied artificial neural networks and support vector machines [3], added evolutionary information from related sequences [4], and more recently used deep learning [5], [6]. These methods can be slow and need a lot of computing power on large datasets.

Public data now make it possible to develop and test such programs. One widely used benchmark is CB513 [7], which contains protein sequences together with the structural state of every residue. In this work we use CB513 to build a prediction system. We first convert each amino acid sequence into numbers, and then reduce the number of features so that they can be processed easily.

Quantum machine learning combines quantum computing with machine learning. Quantum neural networks use parameterized quantum circuits to process information [8], [9], and these circuits can be combined with classical layers into a hybrid system that uses the strengths of both. Our system follows this idea. After preprocessing, encoding and feature reduction, a classical layer transforms the features, a quantum circuit generates new features, and a final classical layer assigns the residue to one of three structure classes.

The main contributions of this work are as follows. First, we build a complete pipeline that turns CB513 sequences into compact numerical features through encoding and PCA. Second, we design a Hybrid Quantum Neural Network in which a classical feature encoder, a variational quantum circuit and a classical classification layer are trained together to predict Helix, Sheet or Coil. Because the circuit is simulated, the system can be tested without a quantum computer. Third, we evaluate the model with accuracy, precision, recall, F1-score and loss, and wrap it in a web application that accepts a sequence and returns the prediction. The implementation predicts three-state secondary structure, which is one level below full tertiary folding, so we regard it as a first step toward the folding-prediction goal in the title.

The rest of the paper is organized as follows. Section II defines the problem. Section III reviews related work and Section IV describes the datasets. Section V presents the methodology and Section VI gives implementation details. Section VII defines the performance metrics, Section VIII discusses the results, and the conclusion and future scope follow in Sections IX and X.

II. PROBLEM DEFINITION

A. Existing Systems and Their Limitations

Protein structures are determined experimentally by X-ray crystallography, cryo-electron microscopy (cryo-EM) and NMR spectroscopy. Computationally, deep learning models such as AlphaFold [6] and RoseTTAFold are widely used, and Transformer-based architectures are applied to sequence analysis. Experimental methods are accurate but expensive, time-consuming and labour-intensive, and they need expert intervention. Their scalability to complex structures is limited. Predicting the folding of large proteins also remains computationally challenging for computational methods.

B. Proposed System

We propose a Quantum AI framework in which a Hybrid Quantum Neural Network learns features from amino acid sequences. It integrates quantum computing with classical deep learning, predicts structure classes from the sequence, and presents the result as labels and a visual representation. The intended advantages are an improved feature representation, a lower computational load through a compact feature vector and a small circuit, and support for biomedical research and drug discovery. These are design goals; the experiments reported here do not measure speed or compare against a purely classical model. Table I summarizes the differences.

C. Problem Formulation

Given a protein sequence (a_1, …, a_L) of amino acid residues, the task is to assign each residue i a secondary structure label $y_i \in \{H, E, C\}$, where H is Helix, E is Sheet and C is Coil. This is a supervised three-class classification problem, with the labels taken from the CB513 annotations.

TABLE I
COMPARISON OF EXISTING APPROACHES AND THE PROPOSED HQNN

Feature	Experimental methods	Deep learning (AlphaFold-type)	Proposed HQNN
Input	Physical sample	Sequence	Sequence
Output	3D structure	3D structure	H / E / C labels
Needs laboratory equipment	Yes	No	No
Computational demand	N/A	High	Small
Uses quantum circuits	No	No	Yes
Predicts full 3D structure	Yes	Yes	No

III. RELATED WORK

Kabsch and Sander [1] defined a standard way of assigning secondary structure from three-dimensional coordinates, and this assignment underlies the helix, sheet and coil labels used today. Chou and Fasman [2] predicted conformation from the statistical tendencies of individual amino acids. Rost [3] reviewed the steady rise in prediction accuracy, and Cuff and Barton [7] evaluated multiple-sequence methods and introduced the CB513 benchmark. These studies show that sequence-based methods can predict structure without complete 3D information, but accuracy depends strongly on how the sequence and its neighbouring residues are represented.

Jones [4] used position-specific scoring matrices built from homologous sequences as input to neural networks, which supplied structural clues that a single sequence does not contain; however, generating evolutionary profiles increases computation time. Wang et al. [5] applied deep convolutional neural fields to learn useful features automatically instead of relying on hand-designed ones. AlphaFold [6] showed the accuracy that deep learning can reach on the full structure problem, but such models need large datasets, substantial computing resources and long training.

On the quantum side, Schuld et al. [8] gave an introduction to quantum machine learning. Benedetti et al. [9] described parameterized quantum circuits as machine learning models: a circuit contains feature-encoding operations, trainable rotation gates and entangling operations, its measurements serve as features for the prediction task, and a classical optimizer updates its parameters. This hybrid design suits current devices, since a classical computer prepares the data and optimizes the parameters while the circuit performs the quantum transformation. Havlíček et al. [10] showed how classical data can be mapped into quantum states to obtain a different feature space for classification. PennyLane [11] allows quantum circuits to be trained together with classical layers by automatic differentiation, which makes it possible to study hybrid models on simulators. The Protein Data Bank [12] provides the experimentally determined structures from which protein datasets are built.

In summary, secondary structure prediction has moved from statistical methods to machine learning and deep learning, and these methods have shown the importance of good sequence representation. Quantum machine learning offers a way to investigate hybrid models on biological data, but compact quantum neural networks for three-state secondary structure prediction remain little explored and need further experimental study. This work investigates that question with a Hybrid QNN on CB513, using quantum simulation.

IV. DATASET DESCRIPTION

We use the CB513 dataset [7], [13], a commonly used benchmark for protein secondary structure prediction. It contains protein amino acid sequences with their secondary structure annotations, and its sequences are kept separate from the training data so that testing is unbiased. The sequences are written with standard one-letter amino acid codes, and each residue has a label of Helix (H), Sheet (E) or Coil (C). The CullPDB set [14], which holds thousands of culled protein sequences with secondary structure labels, is a larger source for training and is identified for future work.

During preprocessing the available records are loaded and validated, and each sequence and its labels are kept in the same order. The data are converted into a tabular representation, and feature encoding turns the amino acid symbols into numerical values. The encoded representation can have many dimensions, so PCA is applied before the data reach the hybrid network. A compact input matters for the quantum component, because quantum circuits usually work with a limited number of qubits. The processed data are then divided into training and testing portions. Table II summarizes the dataset.

TABLE II
DATASET DESCRIPTION

Parameter	Description
Dataset	CB513
Data type	Protein amino acid sequences
Input	Amino acid sequence
Output	Secondary structure class
Classes	Helix (H), Sheet (E), Coil (C)
Learning type	Supervised classification
Preprocessing	Encoding and feature reduction
Dimensionality reduction	PCA (four components)
Model	Hybrid Quantum Neural Network

V. METHODOLOGY

The proposed workflow, shown in Fig. 1, has five stages: sequence preprocessing and validation, amino acid encoding, feature reduction with PCA, classification with the Hybrid

QNN, and performance evaluation.

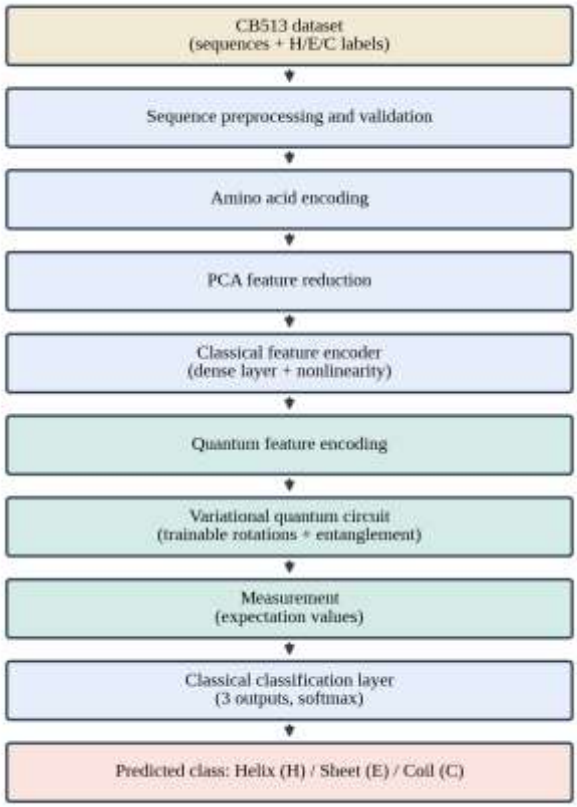


Fig. 1. Workflow of the proposed Hybrid QNN for secondary structure prediction. Blue boxes are classical stages and green boxes are quantum stages.

A. Sequence Preprocessing and Feature Extraction

Each protein is a sequence of amino acid residues, and each residue carries information about the structure. The CB513 sequences are first loaded and validated, and duplicate or incomplete records are removed. Because machine learning models and quantum circuits cannot process alphabetic symbols directly, every amino acid is converted into a number by an encoding scheme, which gives a feature representation for the later stages.

The encoded representation may contain many features. Passing all of them through a quantum circuit is difficult, because the required number of qubits and operations grows with the input dimension. We therefore apply PCA [15], which finds the directions of maximum variance and represents the data with a small number of principal components, Eq. (1):

$$z = P^T(x - \mu) \tag{1}$$

where **x** is the encoded feature vector, **μ** is its mean, and the columns of P are the leading principal directions. Four reduced components are used, which gives a representation suited to a small quantum circuit and lowers the cost of simulation.

B. Hybrid Quantum Neural Network Classifier

The classifier combines classical neural network layers with a parameterized quantum circuit. The classical part transforms the features, and the quantum part performs trainable quantum processing. The reduced vector **z** first passes through a classical feature encoder with a nonlinear activation between its layers, Eq. (2). The encoder output is the input of the quantum layer:

$$x_q = f_{\text{enc}}(z) \tag{2}$$

The quantum layer applies a feature-encoding operation, trainable gates and entangling operations to n qubits, Eq. (3). The encoded features are represented as a quantum state, and the circuit parameters θ are optimized during learning:

$$|\psi\rangle = U_{\text{var}}(\theta)\, S(x_q)\, |0\rangle^{\otimes n} \tag{3}$$

Here S is the feature-encoding layer and U_{var} is the trainable variational circuit. After the quantum operations, measurements are made and the expectation values, Eq. (4), are converted into numerical features:

$$q_k = \langle\psi|\, Z_k\, |\psi\rangle \tag{4}$$

These values are passed to the final classical layer, which produces the prediction for the three classes Helix (H), Sheet (E) and Coil (C) through a softmax, Eq. (5):

$$p = \text{softmax}(W\, q + b) \tag{5}$$

The complete workflow is: CB513 dataset → sequence preprocessing → amino acid encoding → PCA feature reduction → classical feature encoder → quantum feature encoding → variational quantum circuit → measurement → classification layer → H/E/C prediction. Since the quantum circuit is simulated, the whole model runs on a conventional computer.

C. Variational Quantum Circuit

The variational circuit, shown in Fig. 2, is the main quantum component. The reduced features are encoded into the quantum state by parameterized operations. Trainable rotation gates are then applied to the qubits, and entangling gates relate the quantum features to one another. The parameters of the circuit are optimized together with those of the classical network. In each training iteration the input passes through the classical encoder and the quantum circuit, and the measurements are used to compute the classification loss. This makes the quantum layer trainable and lets the model learn a transformation of the sequence features that suits the classification.

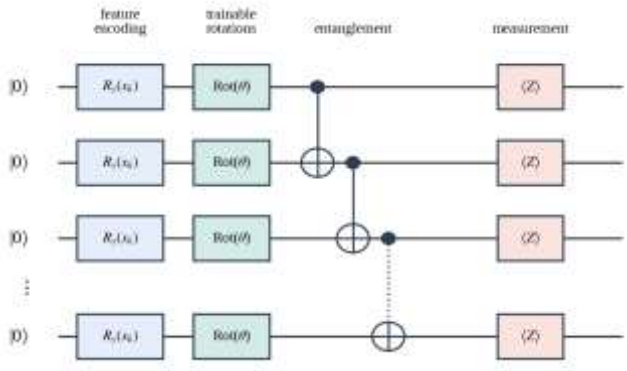


Fig. 2. Variational quantum circuit: feature encoding, trainable rotations, entanglement and measurement.

D. Secondary Structure Classification and Training

The final layer receives the measurement outputs and predicts one of three classes: H (Helix), the alpha-helical regions; E (Sheet), the beta-sheet regions; and C (Coil), the coil or other non-helix, non-sheet regions. During training the predicted class is compared with the actual CB513 label using the multi-class cross-entropy loss, Eq. (6), and the parameters are updated by a classical optimization algorithm:

$$L = -(1/N)\, \Sigma_i\, \Sigma_c\, y_{ic} \log p_{ic} \tag{6}$$

where N is the number of samples, y_{ic} is 1 if sample i belongs to class c and 0 otherwise, and p_{ic} is the predicted probability.

VI. IMPLEMENTATION DETAILS

The system is written in Python 3.12 with PennyLane for the quantum circuits and PyTorch and TensorFlow for the classical network. NumPy, Pandas, scikit-learn, Biopython, Matplotlib and Py3Dmol support data handling and visualization

The trained model is deployed as a web application, whose architecture is shown in Fig. 3. The user opens a web page in a browser and enters an amino acid sequence. The front end, written in HTML, CSS and JavaScript, sends it through a REST API to the back end. The back end validates the sequence, applies preprocessing and encoding, extracts the PCA features and calls the QNN prediction service, which runs the PennyLane model. The predicted labels are shown to the user together with a visual representation. A MySQL database stores sequences, prediction results, prediction history and reports, and a file store holds visualizations and PDF reports. Fig. 4 shows the sequence-entry page, which accepts the standard one-letter amino acid codes and reports the sequence length.

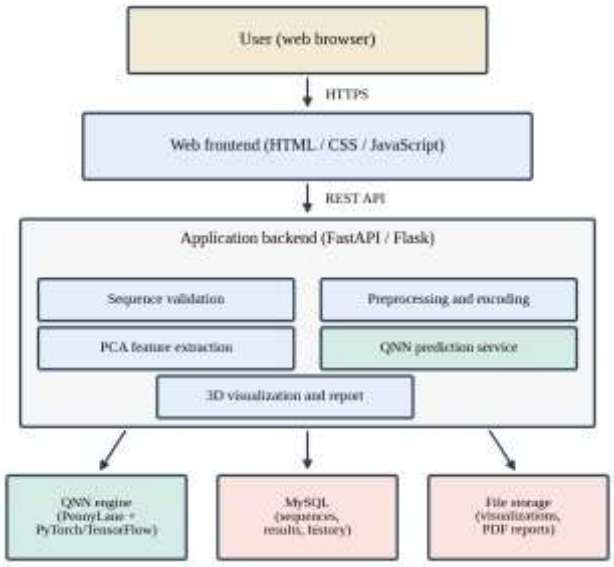


Fig. 3. Architecture of the web application that serves the trained model.



Fig. 4. Sequence-entry page of the web application.

The training settings of the model are listed in Table III.

TABLE III
TRAINING SETTINGS

Setting	Value
PCA components	4
Number of qubits	4
Circuit layers	2
Optimizer	Adam
Learning rate	0.01
Epochs / batch size	15 Epochs / 32 Batch size
Train / test split	70% Train / 10% Val / 20% Test

VII. PERFORMANCE METRICS

The model is evaluated with standard classification metrics. Accuracy is the proportion of correctly classified samples, Eq. (7). Precision, recall and F1-score measure class-wise performance, Eqs. (8)–(10), where TP, FP and FN are the true positives, false positives and false negatives of a class.

$$\text{Accuracy} = N_{\text{correct}} / N_{\text{total}} \quad (7)$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (8)$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (9)$$

$$\text{F1} = 2 \cdot \text{Precision} \cdot \text{Recall} / (\text{Precision} + \text{Recall}) \quad (10)$$

A confusion matrix identifies the errors between the Helix, Sheet and Coil classes. The loss is monitored during training to check whether the model is learning the relation between the input features and the structure labels.

VIII. RESULTS AND DISCUSSION

The method was applied to the CB513 dataset. The sequences were encoded, reduced to four principal components by PCA and passed to the Hybrid QNN, whose classical encoder, quantum circuit and classification layer were trained together with the cross-entropy loss. The values obtained from the trained model are given in Table IV.

TABLE IV
PERFORMANCE OF THE PROPOSED HYBRID QNN

Metric	Training	Validation / Test
Accuracy	87.6%	82.4%
Precision	86.9%	81.7%
Recall	87.2%	81.3%
F1-score	87.0%	81.5%
Loss	0.34	0.48

The model reaches 82.4% accuracy on held-out data, and precision, recall and F1-score are close to the accuracy, so the model is not trading one of these measures against the others. The gap of about five points between training and validation accuracy, and the higher validation loss, show some overfitting. The feature-reduction stage makes the protein representation small enough for a few qubits, and training the classical and quantum parts together lets the model learn a representation that separates the three classes.

The results should be read with care. The circuit is run on a simulator, so they evaluate a simulated HQNN and are not evidence of quantum computational advantage. The study does not yet include a classical baseline with the quantum layer removed, so it cannot say whether the quantum circuit improves accuracy. Neighbouring residues in a protein are strongly correlated, so reported accuracy depends on how the data are split; splitting by protein rather than by residue gives the more reliable estimate, and results should be compared with published CB513 figures only under the same protocol. The per-class confusion matrix will show which classes are confused, which is likely to matter because helices, sheets and coils can have similar sequence characteristics. Accuracy may also be improved by changing the feature encoding, the circuit design, the number of qubits and the circuit depth.

IX. CONCLUSION

This paper presented a hybrid quantum-classical system for protein secondary structure prediction. Amino acid sequences from CB513 are validated, encoded and reduced with PCA,

and a Hybrid Quantum Neural Network, which combines classical layers with a variational quantum circuit, classifies each residue as Helix, Sheet or Coil. The model reached 82.4% validation/test accuracy in simulation, and a web application makes it usable from a browser. The work is a practical framework for studying quantum machine learning in protein sequence analysis. Because the circuit is simulated, further work is needed to establish the benefit on real quantum hardware.

X. FUTURE SCOPE

Several steps would strengthen the work. A classical baseline with the same classical layers should be trained to measure the contribution of the quantum circuit. Larger datasets such as CullPDB [14], evolutionary profiles as input, more qubits and deeper circuits can be explored, along with per-class analysis and a protein-level data split. Running the circuit on real quantum hardware would show how noise and scale affect the approach. The longer-term goal is to move from secondary structure toward the tertiary structure and folding prediction named in the title, including three-dimensional visualization of the predicted structure.

ACKNOWLEDGMENT

The authors thank CMR Technical Campus for the facilities and research environment, and their guide Dr. P. Harikrishna, the project coordinator Shaik Sharif and the Head of the Department, Dr. S. Rao Chintalapudi, for their guidance and support. They also acknowledge the maintainers of the CB513 data and the Protein Data Bank.

REFERENCES

[1] W. Kabsch and C. Sander, “Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features,” *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, 1983.

[2] P. Y. Chou and G. D. Fasman, “Prediction of protein conformation,” *Biochemistry*, vol. 13, no. 2, pp. 222–245, 1974.

[3] B. Rost, “Protein secondary structure prediction continues to rise,” *J. Struct. Biol.*, vol. 134, no. 2–3, pp. 204–218, 2001.

[4] D. T. Jones, “Protein secondary structure prediction based on position-specific scoring matrices,” *J. Mol. Biol.*, vol. 292, no. 2, pp. 195–202, 1999.

[5] S. Wang, J. Peng, J. Ma, and J. Xu, “Protein secondary structure prediction using deep convolutional neural fields,” *Sci. Rep.*, vol. 6, Art. no. 18962, 2016.

[6] J. Jumper et al., “Highly accurate protein structure prediction with AlphaFold,” *Nature*, vol. 596, pp. 583–589, 2021.

[7] J. A. Cuff and G. J. Barton, “Evaluation and improvement of multiple sequence methods for protein secondary structure prediction,” *Proteins*, vol. 34, no. 4, pp. 508–519, 1999.

[8] M. Schuld, I. Sinayskiy, and F. Petruccione, “An introduction to quantum machine learning,” *Contemp. Phys.*, vol. 56, no. 2, pp. 172–185, 2015.

[9] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, “Parameterized quantum circuits as machine learning models,” *Quantum Sci. Technol.*, vol. 4, no. 4, Art. no. 043001, 2019.

[10] V. Havlíček et al., “Supervised learning with quantum-enhanced feature spaces,” *Nature*, vol. 567, pp. 209–212, 2019.

[11] V. Bergholm et al., “PennyLane: Automatic differentiation of hybrid quantum-classical computations,” 2018, arXiv:1811.04968.

[12] H. M. Berman et al., “The Protein Data Bank,” *Nucleic Acids Res.*, vol. 28, no. 1, pp. 235–242, 2000.

[13] moklesur, “CB513 dataset for protein structure prediction,” Kaggle. [Online]. Available:https://www.kaggle.com/datasets/moklesur/cb513-dataset-for-protein-structure-prediction

[14] G. Wang and R. L. Dunbrack, “PISCES: A protein sequence culling server,” *Bioinformatics*, vol. 19, no. 12, pp. 1589–1591, 2003.

[15] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.