Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Matched-Control Benchmarking of Quantum Machine Learning Models for Hierarchical Protein Function Prediction

Farzad Majidi

Programming.Majidi@gmail.com

Islamic Azad University

Research Article

Keywords: protein function prediction, Gene Ontology, quantum machine learning, quantum kernels, matched controls, exponential concentration

Posted Date: November 3rd, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11188049/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Matched-Control Benchmarking of Quantum Machine Learning Models for Hierarchical Protein Function Prediction

Farzad Majidi[1*]
[1] Department of Software Engineering, Islamic Azad University, Sanandaj Branch, Sanandaj, Iran
* Corresponding author's Email: Programming.Majidi@gmail.com

Abstract

Predicting protein function requires assigning hierarchical Gene Ontology (GO) terms to sequences with limited experimental annotation. We tested whether quantum machine learning improves this task when each model is compared with a classical control matched to its architecture or input representation. Five quantum models spanning variational circuits, quantum features, and fidelity kernels were evaluated on 40,000 proteins from the CAFA6 training release, using frozen 480-dimensional ESM-2 embeddings, 180 GO terms, and an internal cluster-disjoint train–validation–test split. Kernel bandwidths were selected through a training-only concentration screen (Gate-0), which also estimated shots required per kernel entry. Across five seeds, the standalone quantum models scored below their matched controls by 0.0023–0.0847 in aspect-mean maximum F-measure; the 48-layer re-uploading kernel was admitted in only two seeds and lost in both. Freezing the variational circuit did not reduce performance. Late fusion of quantum and classical predictions improved on cosine nearest neighbours by 0.0034, but kernel-level mixing selected a purely classical weight, and the fused score (0.6697) remained below that of a classical RBF nearest-neighbour model (0.6719). Finite-shot simulations preserved this ordering, although the quantum kernel ridge model was sensitive to shot noise. These results provide no evidence of a standalone quantum advantage on this benchmark. The fusion gain is consistent with complementary prediction errors, but its mechanism remains to be tested against a matched classical ensemble.

Keywords: protein function prediction; Gene Ontology; quantum machine learning; quantum kernels; matched controls; exponential concentration

1. Introduction

Gene Ontology (GO) annotation associates protein sequences with molecular functions, biological processes and cellular components arranged in a hierarchy (Ashburner et al. 2000). Experimental evidence covers only a small fraction of known sequences, making computational annotation valuable but difficult to validate (Radivojac et al. 2013; Jiang et al. 2016; Zhou et al. 2019; Ramola et al. 2026). A predicted child term entails its ancestors, while missing labels do not necessarily denote absent functions (Clark and Radivojac 2013; Kulmanov et al. 2024a). The benchmark must therefore keep the ontology, split and scoring procedure fixed across models. Its results also need to be read within the chosen label subset rather than as a score for the prospective CAFA challenge.

Protein function prediction has moved from homology transfer to learned representations and ontology-aware models. BLAST-family searches and neighbour voting can exploit annotated homologues (Altschul et al. 1997; Liu et al. 2020), while ProLanGO and DeepGOPlus use neural models and sequence similarity in different combinations (Cao et al. 2017; Kulmanov and Hoehndorf 2020). More recent systems combine protein language models with label relationships, homology or external evidence (Rives et al. 2021; Lin et al. 2023; Zhu et al. 2022; Yuan et al. 2023; Liu et al. 2024). Their inputs and training pipelines differ substantially. For a quantum-versus-classical comparison, the first question is narrower: whether a quantum component improves predictions when both arms use the same proteins, labels and evaluation code.

Quantum embeddings offer nonlinear feature maps for supervised learning (Havlíček et al. 2019; Schuld and Killoran 2019; Jerbi et al. 2023), but apparent improvements can arise from other changes to a pipeline. Expanding an input into many expectation values, changing feature dimension or modifying the classical head can each change accuracy without isolating the quantum state space (Rahimi and Recht 2007). Moreover, trainability and kernel concentration constrain quantum models, and several families admit useful classical surrogates (Larocca et al. 2025; Thanasilp et al. 2024; Schreiber et al. 2023). Our earlier, related preprint tested one dimension-matched quantum feature map on the same internal CAFA6 subset and found that representation width explained most of its improvement (Majidi 2026). The present study broadens that analysis to five standalone quantum models from three feature-map families, gives each a model-specific classical control, evaluates full-dimensional classical baselines, screens kernel concentration and simulates finite-shot estimation. Because the dataset is shared, these experiments extend an existing line of work rather than constituting independent biological validation.

We supply that measurement. We pair each quantum model with a classical control that matches the main architectural or input properties specified for that model while changing the representation under test. We admit kernel

configurations only after a concentration screen fixed on training data, and we report seed-wise paired effects and, for the seed-42 analysis, paired bootstrap intervals. The main contributions of this work are:

- **Contribution 1:** We evaluate five quantum models spanning three feature-map families — a variational circuit, a covariant ZZ feature map and a 48-layer data re-uploading map — on the same hierarchical GO prediction task, under one scoring implementation and one audited cluster-disjoint split.
- **Contribution 2:** We formalise a matching criterion and instantiate one classical control per quantum model, matched approximately on parameter count for the variational model, and on input features and prediction pipeline for the kernel models, and on output width for the learned quantum features.
- **Contribution 3:** We introduce Gate-0, an exponential-concentration screen that selects the kernel bandwidth on training data, rejects configurations whose kernel spread approaches the 2^{-n} floor or whose alignment falls below a classical reference, and converts every candidate into the shot budget a hardware execution would require; a finite-shot simulation then measures what those budgets mean for each downstream model.
- **Contribution 4:** We report, over five seeds, that each standalone quantum model loses in every seed in which Gate-0 admits it, that training the circuit adds nothing over freezing it, that a separately tuned regulariser widens rather than narrows the kernel deficit, and that Gate-0 rejects the 48-layer re-uploading kernel outright in three of five seeds.
- **Contribution 5:** We isolate the single configuration that gains — late fusion of quantum and classical predictions, $+0.0034 \pm 0.0006$ over five seeds — and find that the contrast between kernel-level and score-level fusion is consistent with an ensemble contribution, rather than improved kernel similarity, that it does not reach a plain RBF kernel, and that it inherits a shot budget of order 10^5 per kernel entry.

The remainder of this paper is organized as follows. Section 2 reviews related work and states the research gap. Section 3 presents the theoretical background, the problem formulation, the benchmark architecture, its mathematical model and its algorithms. Section 4 specifies the experimental setup and reports the quantitative comparison, the ablation study, the complexity analysis and the visualization analysis. Section 5 discusses the results, states the limitations and concludes.

2. Related Work

Sequence-based transfer remains a practical reference because annotated neighbours can supply interpretable evidence (Altschul et al. 1997). GODoc combines sequence profiles with neighbour voting (Liu et al. 2020); ProLanGO instead treats annotation as a translation task (Cao et al. 2017), and DeepGOPlus integrates a convolutional predictor with a similarity channel (Kulmanov and Hoehndorf 2020). These systems differ in their dependence on nearby annotated proteins. They contextualise the GO task here, although reproducing their full input pipelines is outside this matched-control benchmark.

Protein language models provide representations for newer ontology-aware predictors (Rives et al. 2021; Lin et al. 2023). TALE, ATGO, SPROF-GO, DeepGOZero, DeepGO-SE, PU-GO and InterLabelGO+ use distinct combinations of label structure, embedding learning, similarity or auxiliary evidence (Cao and Shen 2021; Zhu et al. 2022; Yuan et al. 2023; Kulmanov and Hoehndorf 2022; Kulmanov et al. 2024a; Kulmanov et al. 2024b; Liu et al. 2024). CAFA5 results further illustrate the value of literature and multi-model evidence (Yan et al. 2024; Chervov et al. 2024). We include simplified ATGO and InterLabelGO+ implementations as learned references, and label them as such because the available cached embedding cannot reproduce every input of the published systems.

Previous quantum protein studies have chiefly addressed structure or energy-based tasks, including peptide folding and binding-site geometry (Robert et al. 2021; Doga et al. 2024; Zhang et al. 2026; Zhang et al. 2025). Quantum machine learning has also been surveyed in medical image analysis (Wei et al. 2023). Our related preprint addressed hierarchical GO annotation with one quantum feature map (Majidi 2026). This paper instead asks how several quantum model families compare when each has a corresponding classical control on the same annotation task.

Three lines of theory constrain any design in this space. Gradients of expressive circuits concentrate exponentially in qubit number, an effect now understood through Lie-algebraic and adjoint-representation analyses (Larocca et al. 2025; McClean et al. 2018; Ragone et al. 2024; Fontana et al. 2024; Holmes et al. 2022); the optimisation landscape additionally contains exponentially many traps (Anschuetz and Kiani 2022), and noise induces plateaus of its own (Singkanipa and Lidar 2025). Quantum kernel values concentrate exponentially as well, which explains repeated failures of naive quantum kernels and motivates careful bandwidth and embedding choices (Thanasilp et al. 2024; Gil-Fuster et al. 2024; Shaydulin and Wild 2022; Hubregtsen et al. 2022). Finally, efficient classical surrogates exist for broad families of quantum learning models, and the absence of barren plateaus has been linked to classical simulability, so expressiveness alone cannot imply advantage (Schreiber et al. 2023; Jäger and Krems 2023; Cerezo et al. 2025; Glick et al. 2024); analyses of the role of data and of the prospects of the field reach the same conclusion (Huang et al. 2021; Cerezo et al. 2022). The closest empirical precedent benchmarks generic quantum models against classical ones on small binary datasets and finds that the classical models generally win (Bowles et al. 2024). Table 1 summarises the representative studies of the 2022–2026 period along the dimensions that matter for the present comparison.

Table 1. *Comparison of representative studies, 2022–2026. Reported outcomes are those stated by the cited authors.*

Reference	Year	Method	Dataset	Advantage	Limitation
(Kulmanov and Hoehndorf 2022)	2022	DeepGOZero: ontology axioms as n-balls with a neural predictor	UniProtKB/SwissProt, GO DAG	Predicts terms with no training protein	Sequence-only; weak on BP
(Zhu et al. 2022)	2022	ATGO: language-model embeddings in a triplet network	1,068 benchmark proteins; 3,328 CAFA3 targets	Aligns feature and functional similarity	Requires a large pre-trained model
(Yuan et al. 2023)	2023	SPROF-GO: ProtT5 embeddings, attention, hierarchy, label diffusion	SwissProt time split	Fast inference; enforces hierarchy	Depends on a very large frozen model
(Kulmanov et al. 2024b)	2024	DeepGO-SE: semantic entailment over GO	UniProtKB with GO axioms	Approximates entailment, not only correlation	Higher inference complexity
(Kulmanov et al. 2024a)	2024	PU-GO: positive-unlabelled ranking with ontology priors	Time-based UniProt split	Treats missing annotation as unlabelled	Sensitive to prior specification
(Chervov et al. 2024)	2024	ProtBoost: Py-Boost on ProtT5 with a GNN stack	CAFA5, 142,246 proteins	Simple, fast, reproducible	Tied to one language model
(Yan et al. 2024)	2024	GORetriever: literature-driven retrieval reranking	CAFA5	Uses evidence sequence cannot supply	Useless for uncharacterised proteins
(Liu et al. 2024)	2024	InterLabelGO+: inter-label-aware loss with alignment evidence	CAFA5	Exploits label dependency directly	Needs raw sequences for alignment
(De Paolis Kaluza et al. 2026)	2026	CAFA5 community assessment	CAFA5 prospective benchmark	Independent, prospective, large scale	Partial annotation exploited poorly
(Zhang et al. 2026)	2026	Variational binding-site geometry on utility-level processors	30 fragments, 127 qubits	Hardware-validated end to end	Energy minimisation, not classification
(Zhang et al. 2025)	2025	Hybrid quantum–AI structure prediction on NISQ devices	Protein fragments	Practical NISQ workflow	Structure, not function
(Wei et al. 2023)	2023	Survey of quantum machine learning in medical imaging	Multiple imaging benchmarks	Maps applied QML in biomedicine	Small label spaces; no hierarchy
(Bowles et al. 2024)	2024	Benchmark of quantum models against classical baselines	160 small binary datasets	Large-scale, systematic comparison	Toy tasks; no hierarchical output
(Thanasilp et al. 2024)	2024	Exponential concentration in quantum kernel methods	Theory	Explains failures of naive quantum kernels	Predicts large-n kernels will not work
(Cerezo et al. 2025)	2025	Provable absence of barren plateaus and classical simulability	Theory	Links trainability to simulability	Narrows the space for advantage claims
(Glick et al. 2024)	2024	Covariant quantum kernels for data with group structure	Theory and experiment	Principled embedding for group data	Requires a known symmetry group

Three unresolved problems follow from Table 1, and the present benchmark addresses each directly. First, no study evaluates several quantum model families on hierarchical GO prediction under one scoring implementation, so family-level conclusions rest on comparisons across papers with different data and metrics. Second, comparators do not always control the architectural properties under test, making a reported difference difficult to attribute; the systematic benchmark closest to ours (Bowles et al. 2024) uses generic classical baselines rather than per-model matched controls. Third, kernel concentration is treated as a post-hoc explanation for poor results rather than as an admission criterion applied before a configuration is trained, with the consequence that the shot budget implied by a reported kernel is almost never stated.

3. Proposed Methodology

3.1 Theoretical Background and Problem Formulation

Let $\mathcal{S}$ denote the space of amino-acid sequences and let $\mathcal{G} = (V_{\mathrm{GO}}, E_{\mathrm{GO}})$ be the ontology, a directed acyclic graph whose edges (p, c) encode that term c is a child of term p. We select a term subset $G' = \{g_1, \dots, g_M\} \subset V_{\mathrm{GO}}$ and, for a corpus of N proteins, write

$$X \in \mathbb{R}^{N \times D}, \qquad Y \in \{0,1\}^{N \times M}, \tag{1}$$

where row x_i of X is the frozen protein-language-model embedding of sequence S_i in $D = 480$ dimensions and row y_i of Y is its binary target vector over $M = 180$ terms. Annotation obeys the true-path rule, which we state in (2) as a constraint on the data,

$$y_{ic} = 1 \;\Rightarrow\; y_{ip} = 1 \qquad \forall\, (p, c) \in E_{\mathrm{GO}}, \tag{2}$$

and which the predicted term sets must respect. Encoding a feature vector x into a quantum state $|\phi(x)\rangle$ realises a non-linear map into Hilbert space whose induced fidelity kernel is

$$K(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2, \tag{3}$$

the construction on which quantum kernel and quantum linear models rest (Havlíček et al. 2019; Schuld and Killoran 2019; Jerbi et al. 2023). For an n-qubit embedding the off-diagonal values of (3) concentrate towards 2^{-n} as the embedding becomes expressive (Thanasilp et al. 2024), which is the phenomenon Gate-0 is designed to detect before any model is fitted.

We call a comparison between a quantum model Q and a classical model C *matched* when C approximately matches the specified properties of Q listed for its model family, while the quantum and classical representations differ. Writing $\Pi(\cdot)$ for the vector of such properties, we target $\Pi(Q) \approx \Pi(C)$ and attribute the net effect

$$\Delta = \bar{F}_{\max}(Q) - \bar{F}_{\max}(C) \tag{4}$$

as a controlled comparison conditional on those choices, without claiming complete causal identification. The properties requiring a match differ by model family, which is why a single control cannot serve every model:

$$\Pi = \begin{cases} (\text{parameter count, trunk, head, schedule}) & \text{variational models,} \\ (\text{kernel evaluation protocol, input features, landmarks or } k\text{, regressor}) & \text{kernel models,} \\ (\text{output width, downstream classifier, } k) & \text{learned feature maps.} \end{cases} \tag{5}$$

3.2 Problem definition

Input. A protein sequence $S_i \in \mathcal{S}$, its frozen embedding $x_i \in \mathbb{R}^{480}$, the ontology $\mathcal{G}$ and the selected term set G'.

Output. A probability vector $\hat{y}_i \in [0,1]^M$ over G' and, after thresholding, a transitively closed predicted term set; for each quantum model, the net effect Δ of (4) with a paired bootstrap interval.

Optimization objective. CAFA scores each aspect separately, so we select every model on the aspect-mean protein-centric maximum F-measure,

$$\bar{F}_{\max} = \frac{1}{|A|} \sum_{a \in A} \max_{\tau} \frac{2\, \bar{P}_a(\tau)\, \bar{R}_a(\tau)}{\bar{P}_a(\tau) + \bar{R}_a(\tau)}, \qquad A = \{\mathrm{MFO}, \mathrm{BPO}, \mathrm{CCO}\}, \tag{6}$$

computed on the validation split for selection and on the test split, once, for reporting.

Constraints. The benchmark is subject to four constraints: (i) the approximate matching target $\Pi(Q) \approx \Pi(C)$ of (5) for every reported net effect; (ii) the admission constraint that a kernel configuration enters the comparison only after passing Gate-0 on training data; (iii) the leakage constraint that the term vocabulary, the information content, the scalers and every threshold are fitted without access to test annotations; and (iv) a single-touch constraint under which each test prediction is scored once per seed.

3.3 Overall architecture

Figure 1 illustrates the overall architecture of the benchmark. Data flow is strictly feed-forward. A single embedding matrix, a single cluster-disjoint split and a single scoring implementation feed nine classical reference models and five quantum models; each quantum model is paired with the control its matching rule prescribes; kernel models pass through Gate-0 before they are fitted; and the difference within each pair is the only quantity we interpret. Because only the model changes within a pair, the paired differences in Section 4.5 are interpreted in light of the specified controls, with remaining modelling differences acknowledged.

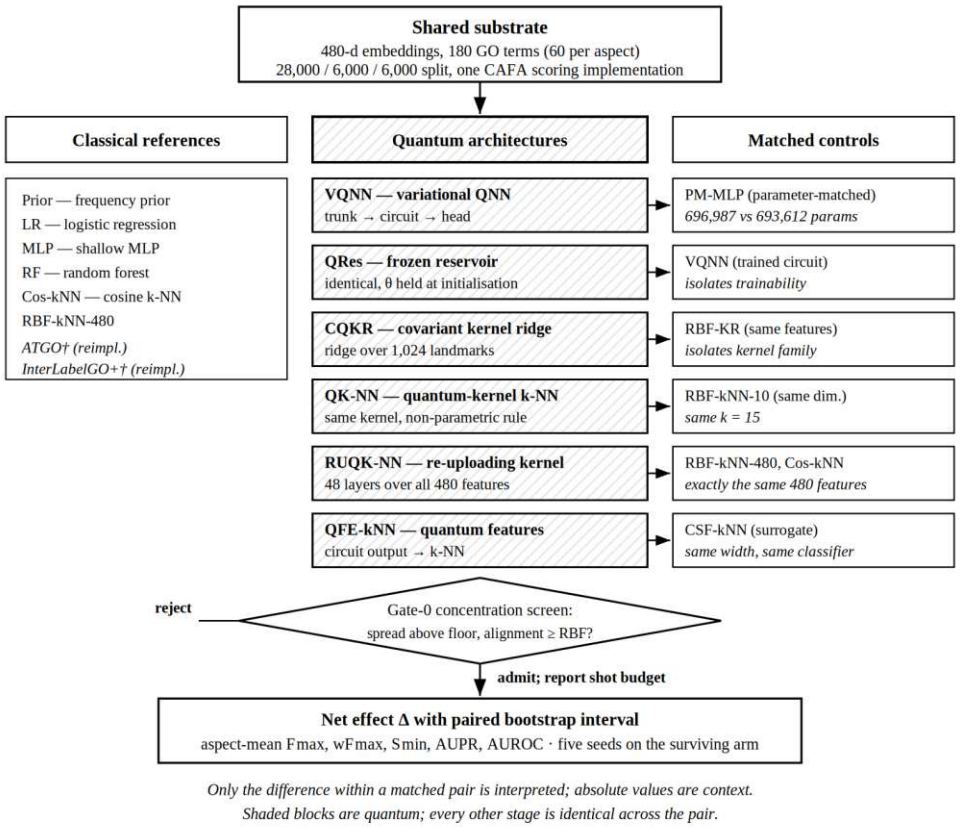


Fig 1. *Overall architecture of the matched-control benchmark. One embedding, one split and one scoring implementation feed nine classical reference models and five quantum models, each paired with a control matched on the property that model is meant to exploit. Shaded blocks are quantum; every other stage is identical within a pair.*

Module 1 — Shared substrate. *Purpose:* to guarantee that every model sees identical data. *Input:* CAFA6 sequences, experimental annotations and the ontology. *Processing:* true-path propagation; term selection and information-content estimation on the training split alone; MinHash clustering over k-mer sets (Broder 1997) combined with exact prefix grouping; assignment of whole clusters to train, validation and test; frozen ESM-2 embedding. *Output:* X, Y, the split and the information-content table.

Module 2 — Quantum feature maps. *Purpose:* to supply three structurally different quantum representations. *Input:* x_i or a reduced view of it. *Processing:* the variational circuit of VQNN, the covariant ZZ feature map of CQKR and QK-NN, and the 48-layer re-uploading map of RUQK-NN, shown in Figure 2 and defined in Section 3.4. *Output:* a vector of 165 expectation values, or a fidelity kernel over landmarks or neighbours.

Module 3 — Matched controls. *Purpose:* to control the main nonquantum design factors within each pair. *Input:* a quantum model and its property vector. *Processing:* construction of the control that (5) prescribes, fitted under the identical seed, schedule, loss and scoring code. *Output:* the control's test predictions.

Module 4 — Gate-0 admission screen. *Purpose:* to prevent a concentrated kernel from entering the comparison and to state its hardware cost before it is fitted. *Input:* 512 training proteins and a bandwidth grid. *Processing:* the three-part admission test of Section 3.4 and the shot budget of each candidate. *Output:* an admitted bandwidth with its shot budget, or a rejection.

Module 5 — Fusion and statistical evaluation. *Purpose:* to test whether a quantum model contributes inside a larger system, and to separate the possible mechanisms. *Input:* quantum and classical scores and kernels. *Processing:* kernel-level mixing and two score-level fusion rules; paired bootstrap over proteins; repetition over seeds. *Output:* net effects with intervals and seed dispersion.

Three design decisions correct specific failure modes, and we state them because each is easy to get wrong. First, controls are matched per model rather than shared, since a control that suits a variational model — same parameter budget — says nothing about a kernel model, whose relevant controls are the input features and downstream prediction pipeline. Second, Gate-0 fixes the bandwidth before any test-set evaluation, so the reported kernels were not tuned on the quantity they are judged by. Third, the test split is touched once per seed, and every threshold, mixing weight and confidence boundary is fitted on validation. Figure 2 shows the three feature-map families in their executed form.

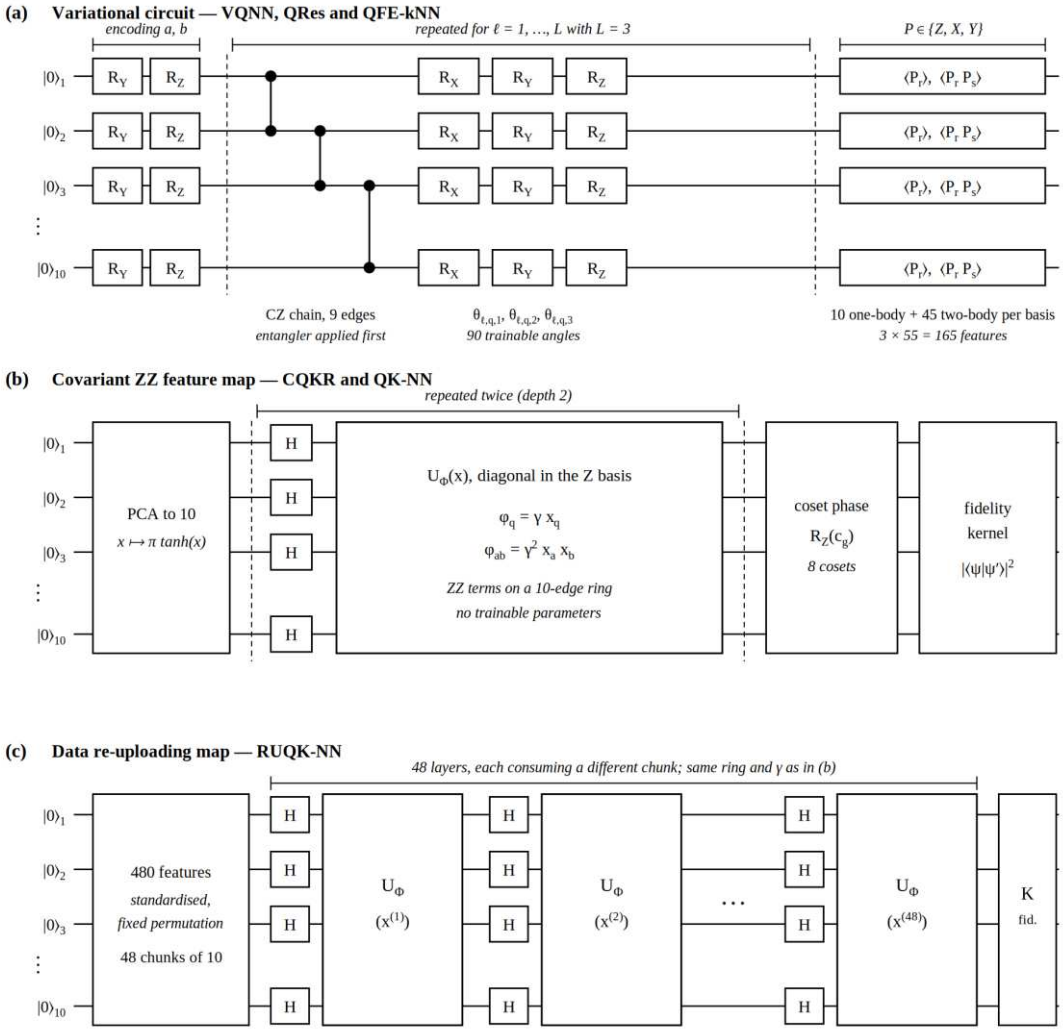


Fig 2. *The three quantum feature-map families, drawn from the executed code. (a) Variational circuit used by VQNN, QRes and QFE-kNN: angle encoding, three layers in which the CZ chain precedes the rotation layer, and one- and two-body read-out in three bases. (b) Covariant ZZ feature map used by CQKR and QK-NN. (c) Data re-uploading map used by RUQK-NN, in which each of 48 layers consumes a different 10-feature chunk.*

3.4 Mathematical model

Variational encoding. For each protein, a classical trunk emits $2n$ angles (a, b) that enter the $n = 10$ qubits as

$$U_{\mathrm{enc}}(a, b) = \prod_{q=1}^{n} R_Z\left(b_q\right) R_Y\left(a_q\right), \tag{7}$$

with R_Y applied first. Each of the $L = 3$ variational layers applies the CZ entangler over the nine edges of a linear chain *before* the single-qubit rotations,

$$U_\ell(\theta) = \left[\prod_{q=1}^{n} R_Z\left(\theta_{\ell,q,3}\right) R_Y\left(\theta_{\ell,q,2}\right) R_X\left(\theta_{\ell,q,1}\right)\right]\left[\prod_{(r,s)\in E_{\mathrm{chain}}} C\,Z_{r,s}\right], \tag{8}$$

so the circuit carries 27 two-qubit gates and 90 trainable angles. **Read-out.** For every basis $P \in \{Z, X, Y\}$ we read the ten one-body and forty-five two-body expectation values,

$$z = \left[\langle P_r\rangle_{r=1}^{n},\ \langle P_r P_s\rangle_{r<s}\right]_{P\in\{Z,X,Y\}} \in \mathbb{R}^{165}, \tag{9}$$

which the classical head of VQNN consumes.

Covariant feature map. CQKR and QK-NN reduce x to ten principal components fitted on the training split, squash each to $\pi\tanh\left(x_q\right)$, and prepare

$$\left|\psi(x)\right\rangle = R_Z\left(c_g\right)\left[U_\Phi(x)\, H^{\otimes n}\right]^2 \left|0\right\rangle^{\otimes n}, \tag{10}$$

where $R_Z\left(c_g\right)$ applies the fixed random phase of the coset g to which k-means on the training split assigns x, one of eight, and $U_\Phi(x)$ is diagonal in the computational basis,

$$U_\Phi(x)\,|k\rangle = \exp\left(i\left[\sum_q \gamma\; x_q\, z_q(k) + \sum_{(a,b)\in E_{\text{ring}}} \gamma^2\, x_a x_b\, z_a(k)\, z_b(k)\right]\right)|k\rangle, \tag{11}$$

with $z_q(k) = \pm 1$ the Z eigenvalue of qubit q in basis state k, E_{ring} the ten edges of a ring, and γ the bandwidth that Gate-0 selects. The map has no trainable parameters.

Re-uploading map. RUQK-NN standardises all 480 features, applies one fixed permutation, and splits the result into 48 chunks $x^{(1)}, \ldots, x^{(48)}$ of ten features, each consumed by its own layer,

$$|\psi(x)\rangle = \prod_{l=1}^{48} U_\Phi\left(x^{(l)}\right) H^{\otimes n}\,|0\rangle^{\otimes n}, \tag{12}$$

with the same ring and the same form of U_Φ as in (11) but without coset phases, so that no second untested factor enters the comparison. All 480 dimensions therefore pass through the circuit, rather than a ten-dimensional projection.

Kernel ridge regression. CQKR fits a Nyström ridge regressor over $m = 1{,}024$ training landmarks. With K_{nm} the kernel between training proteins and landmarks and K_{mm} the kernel among landmarks,

$$Z = K_{nm} K_{mm}^{-1/2}, \qquad W = (Z^\top Z + \lambda I)^{-1} Z^\top Y, \qquad \hat{Y} = \text{clip}\big(K_{*m} K_{mm}^{-1/2} W,\, 0,\, 1\big), \tag{13}$$

and RBF-KR applies the identical pipeline to an RBF kernel with a median-heuristic bandwidth on the same ten features.

Nearest-neighbour scoring. Every nearest-neighbour model, quantum or classical, scores a query as a similarity-weighted average over its $k = 15$ most similar training proteins,

$$\hat{y}(x) = \frac{\sum_{j\in\mathcal{N}_k(x)} w_j\, y_j}{\sum_{j\in\mathcal{N}_k(x)} w_j}, \qquad w_j = \max\big(0,\, s(x, x_j)\big), \tag{14}$$

where s is the quantum fidelity kernel for QK-NN and RUQK-NN, cosine similarity for Cos-kNN, and the RBF kernel for RBF-kNN-10 and RBF-kNN-480.

Fusion. Kernel-level mixing, K-Mix, combines the two similarity matrices before any prediction is made,

$$K_\alpha = \alpha\, K_{\cos} + (1-\alpha)\, K_q, \tag{15}$$

whereas quantum–classical late fusion, QCF, combines two independently produced score matrices with a single weight,

$$\hat{y}_{\text{QCF}} = \alpha_g\, \hat{y}_{\cos} + \big(1-\alpha_g\big)\, \hat{y}_q. \tag{16}$$

Confidence-gated fusion, CG-QCF, lets the weight depend on the classical predictor's confidence, measured by the margin μ between the similarity of the nearest and of the k-th neighbour. With quartile boundaries of μ fitted on validation and $b(i)$ the quartile of protein i,

$$\hat{y}_{\text{CG},i} = \alpha_{b(i)}\, \hat{y}_{\cos,i} + \big(1-\alpha_{b(i)}\big)\hat{y}_{q,i}, \qquad b(i) \in \{1,2,3,4\}. \tag{17}$$

Gate-0. On $m = 512$ training proteins, Gate-0 measures kernel–target alignment with centred matrices,

$$\text{KTA}(K) = \frac{\langle \tilde{K}, \tilde{T}\rangle_F}{\|\tilde{K}\|_F\, \|\tilde{T}\|_F}, \qquad T = YY^\top, \quad \tilde{A} = A - \bar{A}, \tag{18}$$

and admits a bandwidth γ only if all three conditions hold,

$$\sigma_K \geq \sigma_{\min}, \qquad \sigma_K \geq 2^{-n}, \qquad \text{KTA}\big(K_q\big) \geq \rho_{\min}\, |\text{KTA}(K_{\text{RBF}})|, \tag{19}$$

with σ_K the standard deviation of the off-diagonal kernel entries, $\sigma_{\min} = 0.010$ and $\rho_{\min} = 0.80$ fixed before any test evaluation. Among the admitted bandwidths the one with the largest alignment is selected. Since a hardware estimate of each kernel entry carries sampling error $1/\sqrt{M_{\text{shot}}}$, resolving the kernel's spread at five standard errors requires

$$M_{\text{shot}}(\gamma) = \left(\frac{5}{\sigma_K}\right)^2 \tag{20}$$

shots per kernel entry, which Gate-0 reports for every candidate.

3.5 Algorithm and workflow

Figure 3 shows the control flow across the benchmark, including the Gate-0 rejection loop and the dissection branch that is entered only when a quantum model produces a positive interval.

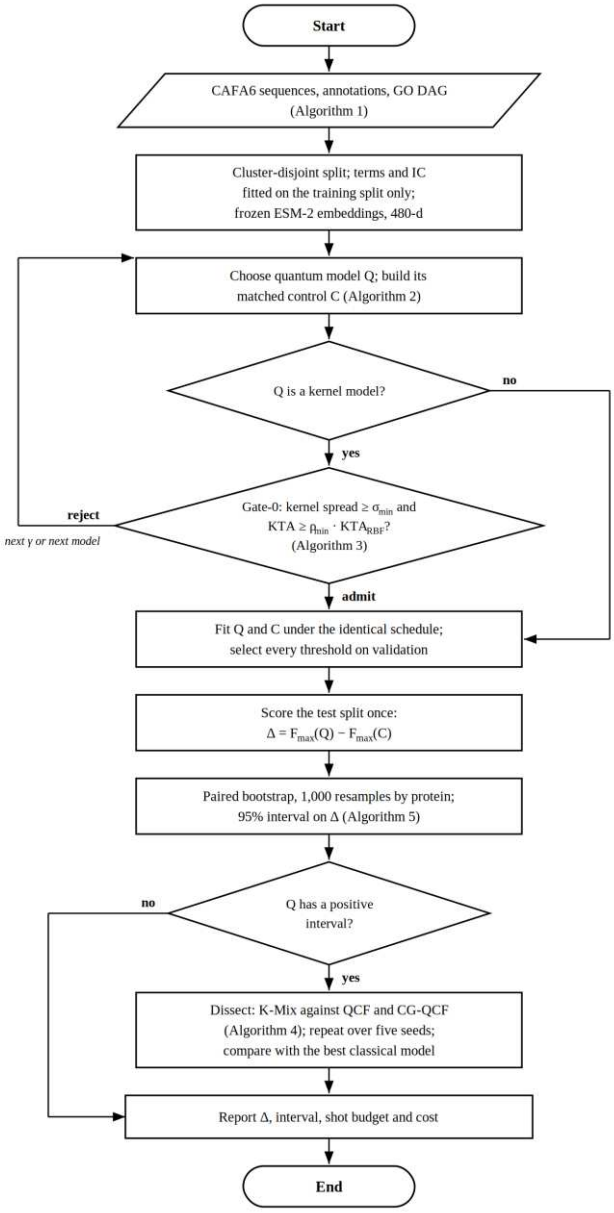


Fig3. *Workflow of the matched-control benchmark, from data preparation through Gate-0, matched fitting and paired evaluation to the dissection of any positive result.*

Algorithm 1: Data preparation and cluster-disjoint splitting

Require: sequences S, annotations A, ontology DAG G, minimum training frequency $f_{min} = 250$, terms per aspect $n_a = 60$, seed 42
Ensure: splits D_{train}, D_{val}, D_{test}; term set G'; targets Y; conditional IC

drop records with a missing sequence or annotation; merge duplicates
remap alternate and obsolete GO identifiers to current primaries
$A' \leftarrow A$
for all $(i, c) \in A$ **do** *» true-path rule*
  $A' \leftarrow A' \cup \{(i, a) : a \in \text{Ancestors}(c, G)\}$
end for
$C \leftarrow$ MinHash clusters over k-mer sets $\cup$ exact prefix groups
assign whole clusters of C to train, val and test *» never split a cluster*
assert no cluster and no identifier is shared by two splits
$G' \leftarrow$ the n_a most frequent terms per aspect in D_{train} *» training only*
$IC(c) \leftarrow -\log \Pr(c \mid \text{Pa}(c))$ on D_{train} annotations *» training only*
embed each sequence with frozen ESM-2; mean over residue tokens

Algorithm 2: Construction of the matched control

Require: quantum model Q, its property vector $\Pi(Q)$, shared features and split
Ensure: classical control C with $\Pi(C) = \Pi(Q)$

if Q is variational **then**
  C $\leftarrow$ Q with the circuit replaced by a classical block *» PM-MLP*
  match trunk, head, schedule and parameter count
else if Q is a kernel model **then**
  C $\leftarrow$ the same pipeline with K_q replaced by an RBF kernel *» RBF-KR, RBF-kNN*
  match input features, landmarks or k, and the regressor
else if Q is a feature map **then**
  C $\leftarrow$ a classical surrogate map of the same output width *» CSF-kNN*
  match the downstream classifier and k
end if
fit Q and C under one seed, schedule, loss and scoring code
return C

Algorithm 3: Gate-0 exponential-concentration screen

Require: training features, grid $\Gamma = \{0.05, 0.10, 0.20, 0.35, 0.60, 1.00\}$, sample size $m = 512$, thresholds $\sigma_{min} = 0.010$, $\rho_{min} = 0.80$
Ensure: admitted bandwidth γ^* and shot budget, or a rejection

draw m training proteins without replacement
$\kappa_{RBF} \leftarrow$ KTA of the RBF kernel on the sample
for all $\gamma \in \Gamma$ **do**
  K $\leftarrow$ fidelity kernel of the quantum map at γ on the sample
  $\sigma \leftarrow$ standard deviation of the off-diagonal entries of K
  $M_{shot}(\gamma) \leftarrow (5 / \sigma)^2$ *» 5σ resolution on hardware*
  admit γ **if** $\sigma \geq \sigma_{min}$ **and** $\sigma \geq 2^{-n}$ **and** $\text{KTA}(K) \geq \rho_{min} |\kappa_{RBF}|$
end for
if no γ is admitted **then return** reject *» a publishable negative*
$\gamma^* \leftarrow$ the admitted γ with the largest KTA *» training data only*
return γ^*, $M_{shot}(\gamma^*)$

Algorithm 4: Kernel-level and score-level quantum–classical fusion

Require: scores $\hat{y}_{cos}$, $\hat{y}_q$; kernels K_{cos}, K_q; coarse grid $A_0 = \{0, 0.25, \ldots, 1\}$; fine grid $A = \{0, 0.05, \ldots, 1\}$; validation split
Ensure: K-Mix, QCF and CG-QCF predictions for the test split

for all $\alpha \in A_0$ **do** score $K_\alpha \leftarrow \alpha K_{cos} + (1 - \alpha) K_q$ on validation **end for**
K-Mix $\leftarrow K_\alpha$ at the best validation α *» combine before prediction*
$\alpha_g \leftarrow \text{argmax}_\alpha F(\alpha \hat{y}_{cos} + (1 - \alpha) \hat{y}_q)$ on validation
QCF $\leftarrow \alpha_g \hat{y}_{cos} + (1 - \alpha_g) \hat{y}_q$ *» combine after prediction*
m $\leftarrow$ similarity to nearest minus similarity to k-th neighbour
fit quartile edges of m on validation; apply them to test *» no leakage*
for b = 1 **to** 4 **do**
    $\alpha_b \leftarrow$ best α within confidence quartile b on validation
end for
CG-QCF $\leftarrow \alpha_{b(i)} \hat{y}_{cos,i} + (1 - \alpha_{b(i)}) \hat{y}_{q,i}$ **for** each protein i
return K-Mix, QCF, CG-QCF

Algorithm 5: Evaluation and paired bootstrap

Require: test scores of Q and C, targets Y_{test}, IC, seeds S, B = 1000
Ensure: net effect Δ, its 95% interval and the seed dispersion

for all seeds $s \in S$ **do**
    repair the hierarchy from leaves to roots in both score matrices
    $\Delta \leftarrow F_{max}(Q) - F_{max}(C)$ *» aspect mean*
    for b = 1 **to** B **do**
        draw N test proteins with replacement, the same draw **for** Q and C
        $\Delta_b \leftarrow F_{max}(Q) - F_{max}(C)$ on the resample
    end for
    report Δ and the 2.5% and 97.5% quantiles of $\{\Delta_b\}$
end for
report the mean and standard deviation of Δ over S
t-test on the seed-wise Δ; Wilcoxon as a robustness check

4. Experimental Setup

4.1 Dataset description

The experiments load a precomputed partition of the CAFA6 training release: 28,000 proteins for training, 6,000 for validation and 6,000 for testing, with 480-dimensional embeddings and 180 GO terms. The supplied analysis notebook verifies these array dimensions and audits the split, but it does not itself regenerate embeddings, labels or cluster assignments from the original sequences. Those upstream steps therefore require the preparation manifests and code described in the data-availability statement. The reported test scores belong to this internal partition and must not be compared with official CAFA leaderboard results.

Table 2. *Dataset summary.*

Dataset	Samples	Features	Task
CAFA6 (internal split, this work)	40,000 proteins (28,000 / 6,000 / 6,000)	480-d ESM-2 embedding	Hierarchical multi-label classification over 180 GO terms

Table 3. *Corpus construction and label statistics.*

Property	Value
Train / validation / test proteins	28,000 / 6,000 / 6,000
GO terms (MFO / BPO / CCO)	180 (60 / 60 / 60)
Term selection	training split only, minimum training frequency 250
Label density (train / val / test)	0.1059 / 0.0969 / 0.0974

Property	Value
Labels per protein (train / val / test)	19.05 / 17.43 / 17.54
Training label density (MF / BP / CC)	0.05438 / 0.14094 / 0.12225
Terms without a positive in any split	0
Benchmark proteins (MF / BP / CC / any)	4,094 / 4,298 / 4,365 / 5,991
Mean conditional information content (all / MF / BP / CC)	3.877 / 4.803 / 3.010 / 3.817
Information-content source	training-split annotations only
Clustering	MinHash over k-mer sets with exact prefix grouping
Clusters over 40,000 proteins	35,803
Clusters or identifiers shared by two splits	0 (all three pairs audited)
Preparation protocol / seed	`prep-v2-training-only-vocabulary` / 42
Aspect assignment	135 terms from the annotation table, 45 from `go-basic.obo`

Three properties of the partition matter for how the results should be read, and we audited each rather than asserting it. The split is disjoint at the level of sequence clusters, not only of individual proteins: 35,803 clusters over the 40,000 proteins, and no cluster contributes proteins to more than one split, which closes the route by which near-duplicate sequences would otherwise inflate held-out accuracy. The term vocabulary is selected from the training split alone, so no test annotation influences which terms are scored. And the conditional information content that weights $wF_{\max}$ and $S_{\min}$ is fitted on training annotations only. Two boundaries on these claims are recorded in the preparation manifest and we state them plainly: the clustering bounds gross near-duplication but does not certify a sequence-identity threshold, and the partition is a retrospective repartition of released material rather than a prospective time-based split.

4.2 Implementation details

Table 4 records the configuration as executed. Optimizer settings were verified against the executed training cell in the accompanying notebook; the preparation of the precomputed input arrays is a separate upstream step.

Table 4. *Implementation configuration, as executed.*

Item	Setting
Compute environment	Kaggle notebook; classical stage on GPU, quantum simulation on a state-vector engine
Python / PyTorch / transformers / NumPy	3.12.13 / 2.10.0+cu128 / 5.0.0 / 2.0.2
Embedding model	`facebook/esm2_t12_35M_UR50D`, revision `6fbf070e`, frozen, 480-d
Pooling / residue limit	mean over residue tokens of the last hidden state, special tokens excluded / 1,022
Simulator verification	fast state-vector engine against PennyLane (Bergholm et al. 2018): max deviation 4.17×10^{-7} (linear/CZ), 6.56×10^{-7} (tree/CZ), 2.98×10^{-7} (ring/CNOT)
VQNN circuit	10 qubits, 3 layers, CZ chain, 27 two-qubit gates, 20 input angles, 90 trainable angles, 165 read-out features
VQNN parameters	trunk 379,136 + angle map 5,140 + head 309,246 + circuit 90 = 693,612
PM-MLP parameters	trunk 379,136 + angle map 5,140 + head 309,246 + classical block 3,465 = 696,987
Circuit initialisation	$\theta \sim 0.02\,\mathcal{N}(0,1)$
VQNN training	20 epochs, best checkpoint on validation $\bar{F}_{\max}$
CQKR / QK-NN map	PCA to 10, $\pi\tanh$ squashing, depth 2, 10-edge ring, 8 cosets, no trainable parameters
CQKR regressor	Nyström, 1,024 landmarks; ridge 1.0 in the main run, tuned per kernel in Section 4.6
RUQK-NN map	480 standardised features, fixed permutation, 48 layers of 10, same ring, no cosets
Gate-0	512 training proteins; grid 0.05, 0.10, 0.20, 0.35, 0.60, 1.00; $\sigma_{\min} = 0.010$; $\rho_{\min} = 0.80$
Nearest-neighbour models	$k = 15$, similarity-weighted average of neighbour labels

Item	Setting
Fusion weights	K-Mix on 0, 0.25, 0.50, 0.75, 1.00; QCF and CG-QCF on 21 points from 0 to 1
Bootstrap / seeds	1,000 paired resamples by protein / 42, 0, 1, 2, 3
Optimizer, learning rate, batch size, weight decay	AdamW; learning rate 2×10^{-3}; batch size 256; weight decay 10^{-5} (VQNN, QRes and PM-MLP); cosine schedule with 5% warm-up; EMA decay 0.995; early-stopping patience 5

AI-assisted manuscript preparation. An OpenAI language model assisted with editorial rewriting of portions of the abstract, introduction, related work and discussion, and with reference formatting. The reported calculations, tables and figures derive from the executed notebook and underlying data. The author is responsible for reviewing the final manuscript and verifying all claims and citations.

4.3 Baseline methods

Table 5 fixes the short names used in every table and figure. Each code identifies one fitted model; the codes CQKR, QK-NN, QFE-kNN, RUQK-NN and CG-QCF are those used in the executed notebook, so every number can be traced to the cell that produced it. We deliberately label no model as "proposed": the contribution of this paper is the matched-control protocol and Gate-0, not a new predictor, and Section 4.6 shows that the best model containing a quantum component does not outperform the classical RBF-kNN-480.

Table 5. *Model codes. Q denotes a quantum model, C its matched control, F a fusion of the two and B a classical reference baseline.*

Code	Role	Model	Matched against
VQNN	Q	variational quantum neural network: trunk, 10-qubit circuit, head	PM-MLP
QRes	Q	VQNN with circuit parameters frozen at initialisation	VQNN
CQKR	Q	covariant quantum kernel ridge regression	RBF-KR
QK-NN	Q	nearest neighbours on the covariant quantum kernel	RBF-kNN-10
RUQK-NN	Q	nearest neighbours on the 48-layer re-uploading kernel	RBF-kNN-480, Cos-kNN
QFE-kNN	Q	nearest neighbours on the 165 quantum features of the trained VQNN	CSF-kNN
PM-MLP	C	classical network matched to VQNN in parameters, trunk, head and schedule	—
RBF-KR	C	RBF kernel ridge regression, same features and landmarks as CQKR	—
RBF-kNN-10	C	RBF nearest neighbours on the ten features QK-NN consumes	—
RBF-kNN-480	C, B	RBF nearest neighbours on the 480 standardised features	—
CSF-kNN	C	nearest neighbours on classical surrogate features of width 165	—
K-Mix	F	kernel-level mixing of the quantum and cosine kernels	Cos-kNN
QCF	F	late fusion of QK-NN and Cos-kNN scores with one global weight	Cos-kNN
CG-QCF	F	QCF with the weight fitted per quartile of classical confidence	Cos-kNN
Cos-kNN	C, B	cosine nearest neighbours on the 480 raw features	—
Prior	B	frequency prior	—
LR	B	multi-label logistic regression	—
MLP	B	shallow multilayer perceptron	—
RF	B	random forest, 30 trees, depth 10	—
ATGO†	B	reimplementation of ATGO (Zhu et al. 2022)	—
InterLabelGO+†	B	reimplementation of InterLabelGO+ (Liu et al. 2024)	—

We compare against a frequency prior, multi-label logistic regression, cosine and RBF nearest neighbours, a random forest and a shallow MLP, and against reimplementations of ATGO (Zhu et al. 2022) and InterLabelGO+ (Liu et al. 2024). Both reimplementations depart from their sources because the cached representation does not carry the required inputs: ATGO uses three hidden layers of a large protein language model where we supply one branch over the 480-dimensional embedding, and its alignment component is not run; InterLabelGO+ uses three parallel networks over three language-model layers where we use one, and its hierarchical post-processing and alignment components are not run, although its inter-label loss is implemented exactly. We therefore read these two rows as strong learned references on our data rather than as reproductions of the published systems, which the dagger marks.

4.4 Evaluation metrics

We report every metric per aspect and as an aspect mean, computed on the transitively closed predictions.

Protein-centric maximum F-measure. With T_i the true and $P_i(\tau)$ the predicted term set of protein i at threshold τ,

$$\bar{P}(\tau) = \frac{1}{|\mathcal{B}_\tau|}\sum_{i\in\mathcal{B}_\tau}\frac{|P_i(\tau)\cap T_i|}{|P_i(\tau)|}, \qquad \bar{R}(\tau) = \frac{1}{N}\sum_{i=1}^{N}\frac{|P_i(\tau)\cap T_i|}{|T_i|}, \tag{21}$$

where $\mathcal{B}_\tau$ is the set of proteins carrying at least one prediction at τ; $F_{\max}$ follows as in (6). It is the CAFA community's primary criterion, but it collapses a whole precision–recall curve onto one operating point. $wF_{\max}$ weights each term by its information content.

Area under the precision–recall curve.

$$\mathrm{AUPR} = \int_0^1 P\,(R)\,dR, \tag{22}$$

computed micro over the flattened $N \times M$ matrix. It is the appropriate summary under extreme class imbalance, because it ignores the true negatives that dominate a 180-term label space.

Semantic distance. With remaining uncertainty ru and misinformation mi,

$$S_{\min} = \min_\tau \sqrt{ru(\tau)^2 + mi(\tau)^2}, \qquad ru(\tau) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c\in T_i\setminus P_i(\tau)} I\,C(c), \qquad mi(\tau) = \frac{1}{N}\sum_{i=1}^{N}\sum_{c\in P_i(\tau)\setminus T_i} I\,C(c), \tag{23}$$

where the information content is conditional on the parents,

$$IC(c) = -\log\Pr\big(c \mid \mathrm{Pa}(c)\big), \tag{24}$$

estimated from training annotations alone. The conditional form in (24) matters: the marginal alternative $-\log\Pr(c)$ re-counts each ancestor's information at every descendant (Clark and Radivojac 2013).

Statistical protocol. We repeat every comparison over five seeds (42, 0, 1, 2, 3), rerunning the whole pipeline of each pair — including Gate-0, whose 512-protein sample depends on the seed — and report the mean and standard deviation of Δ across seeds, the number of seeds in which Δ is negative, a one-sample t-test on the seed-wise differences and the paired effect size $d_z = \bar{\Delta}/s_\Delta$. Within seed 42 we additionally attach a paired bootstrap interval over proteins, resampling test proteins with replacement 1,000 times with the same draw for both models. At $n = 5$ the smallest attainable two-sided Wilcoxon p is 0.0625, so the non-parametric test cannot reach 0.05 by construction; we report the t-test as an exploratory summary and Wilcoxon as a sensitivity check; five seeds and one split limit inferential strength.

4.5 Quantitative comparison

Table 6 establishes where the classical field sits on this data. Cos-kNN reaches 0.6663 aspect-mean $F_{\max}$ in 0.5 s, RBF-kNN-480 reaches 0.6719 in 0.2 s, and the two learned reimplementations reach 0.6786 and 0.6877. Every quantum model must be read against these numbers.

Table 6. *Classical baselines and literature reimplementations. $\bar{F}_{max}$ and $\bar{wF}_{max}$ are aspect means; the last two metric columns are pooled over all 180 terms.*

Method	$\bar{F}_{\max}$	$\bar{wF}_{\max}$	pooled $F_{\max}$	pooled $S_{\min}$	fit time (s)
Prior	0.5348	0.4246	0.4465	38.6974	0.0
LR	0.5365	0.4807	0.4367	36.9334	12.5
MLP	0.5685	0.5075	0.4671	35.4320	14.6
RF	0.6193	0.5368	0.5097	35.0089	45.6
Cos-kNN	0.6663	0.5990	0.5533	32.3032	0.5
RBF-kNN-480	0.6719	0.6060	0.5549	32.1862	0.2
InterLabelGO+†	0.6786	0.6192	0.5596	32.3799	21.3
ATGO†	0.6877	0.6251	0.5727	31.1687	24.6

Table 7 reports each quantum model beside its matched control over five attempted seeds (two admitted seeds for RUQK-NN), and Figure 4 places every method on a common axis. Every pair was rerun end to end in each seed; the seed-42 values reproduce the single-run figures of our earlier analysis to within 4×10^{-5}.

Table 7. *Each quantum model against its matched control over five seeds. Values are mean ± s.d. of aspect-mean F_{max} on the test split; "negative" counts the seeds with $\Delta < 0$; p is from a one-sample t-test on the seed-wise differences. With $n = 5$ and a uniform sign, the two-sided Wilcoxon p is 0.0625 for uniformly signed five-seed rows. The RUQK-NN row is descriptive because only two seeds pass Gate-0.*

Quantum model	Matched control	n	Quantum	Control	Δ	Negative	p	d_z
VQNN	PM-MLP	5	0.5603 ± 0.0013	0.5626 ± 0.0006	−0.0023 ± 0.0015	5 / 5	0.030	−1.47
QRes	VQNN	5	0.5611 ± 0.0008	0.5603 ± 0.0013	+0.0008 ± 0.0016	2 / 5	0.353	+0.47
CQKR	RBF-KR, ridge 1.0	5	0.5927 ± 0.0137	0.6297 ± 0.0019	−0.0370 ± 0.0136	5 / 5	0.004	−2.71
CQKR	RBF-KR, tuned ridge	5	0.5927 ± 0.0137	0.6329 ± 0.0027	−0.0402 ± 0.0139	5 / 5	0.003	−2.89
QK-NN	RBF-kNN-10	5	0.6066 ± 0.0048	0.6252 ± 0.0011	−0.0186 ± 0.0043	5 / 5	< 0.001	−4.35
RUQK-NN	RBF-kNN-480	2	0.6116 ± 0.0004	0.6719 ± 0.0001	−0.0603 ± 0.0005	2 / 2	n/a	—
RUQK-NN	Cos-kNN	2	0.6116 ± 0.0004	0.6663 ± 0.0000	−0.0547 ± 0.0004	2 / 2	n/a	—
QFE-kNN	CSF-kNN	5	0.5776 ± 0.0048	0.6624 ± 0.0010	−0.0847 ± 0.0056	5 / 5	< 0.001	−15.17

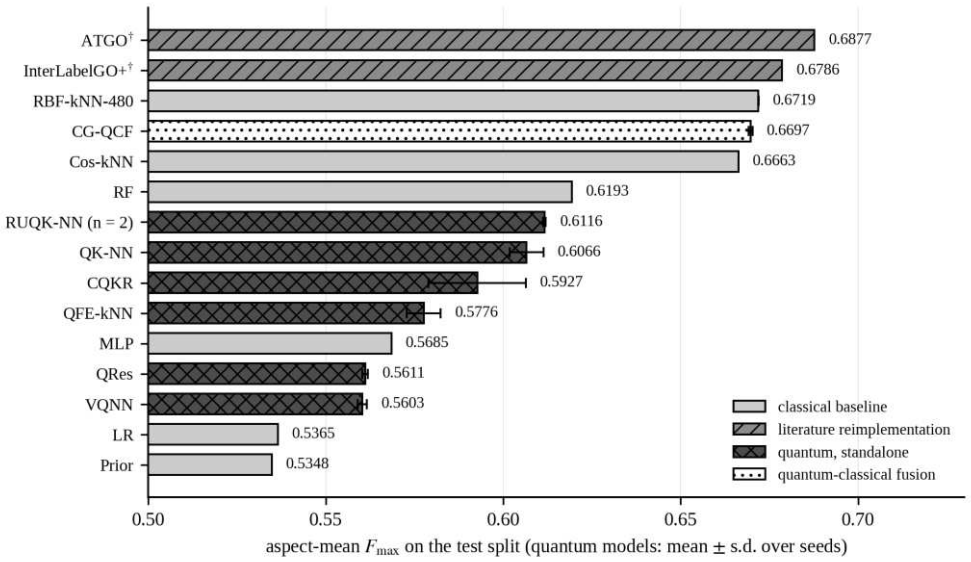


Fig 4. *Aspect-mean F_{max} of every method on the test split; quantum models and CG-QCF show the mean and standard deviation over seeds. No standalone quantum model reaches the cosine nearest-neighbour baseline, and the best model containing a quantum component, CG-QCF, sits below RBF-kNN-480.*

Every standalone quantum model loses to its matched control in each seed in which that model is admitted. The mean deficits range from 0.0023 for VQNN against PM-MLP to 0.0847 for QFE-kNN against CSF-kNN, and the five-seed comparisons have nominal p < 0.05 under the exploratory seed-level t-test, from $p = 0.030$ for the smallest effect to $p < 0.001$ for the largest. The smallest deficit is also the least stable, varying between −0.0038 and −0.0002 across seeds, yet it never changes sign. The only comparison whose interval straddles zero is QRes against VQNN, which is the trainability ablation rather than a quantum-against-classical test. Figure 5 plots the net effects with 95% intervals over seeds. No inferential p-value is interpreted for RUQK-NN with two admitted seeds.

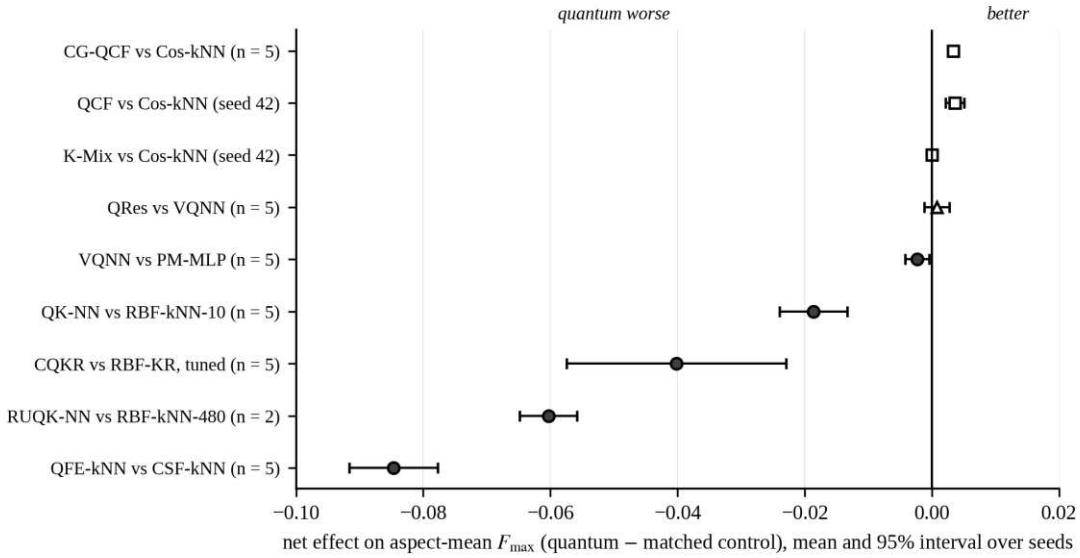


Fig 5. *Net effect of each quantum component against its matched control: mean and 95% t-interval over seeds, with n given for each row; QCF and K-Mix show the seed-42 bootstrap interval. Every quantum-against-classical interval lies entirely left of zero. The open triangle marks the trainability ablation, whose interval contains zero, and the open squares mark the fusion rules.*

4.6 Ablation study

We remove or vary one factor at a time, and the pairs are unusually clean because each differs at a single interface. Three explanations for the negative results are tested directly, and a fourth analysis dissects the one positive result.

Circuit trainability. QRes is VQNN with its circuit frozen at initialisation. Over five seeds the frozen circuit differs from the trained one by $+0.0008 \pm 0.0016$, with the sign split three to two and $p = 0.35$, so the two are indistinguishable. The circuit did train: in seed 42 the parameter displacement $\| \theta - \theta_0 \|$ reaches 9.295×10^{-1}, or 550.4% of its initial norm, when trained and 1.863×10^{-9} when frozen. Optimisation failure therefore does not explain the VQNN result; training the circuit simply buys nothing.

Regularisation. The quantum and classical kernels of CQKR differ enormously in scale: the mean off-diagonal entry is 0.004073 for the quantum kernel against 0.595050 for the RBF kernel, a ratio of 0.0068. A ridge coefficient fixed at the same value for both could therefore impose very different effective regularisation. We swept the coefficient over eight values, selected it on validation separately for each kernel, and evaluated the test split only at the fixed and the selected value. Table 8 reports the sweep. The quantum arm is almost perfectly insensitive to the coefficient, moving by 0.0008 on validation across five orders of magnitude, because the Nyström whitening in (13) already absorbs the scale difference. In seed 42, tuning widens the gap rather than narrowing it: RBF-KR improves from 0.6313 to 0.6357 at its own optimum of 0.1 while CQKR stays at 0.5891 at its optimum of 1.0, moving the net effect from −0.0422 to −0.0466. An independent correction that rescales both kernels to unit mean off-diagonal entry agrees to within 0.0010, at −0.0476, and the analysis reproduces the original quantum arm exactly, at 0.5891. Across five seeds the tuned deficit is -0.0402 ± 0.0139 against -0.0370 ± 0.0136 with the ridge fixed, so tuning widens the gap in every seed. CQKR varies more across seeds than any other quantum model, from 0.5823 to 0.6168, and Table 9 identifies the cause. Gate-0 selects $\gamma = 0.35$ in four seeds but $\gamma = 0.20$ in seed 0, where the two bandwidths align almost equally with the labels (0.1250 against 0.1236). The kernel at $\gamma = 0.20$ has a spread 83.4 times the floor rather than about 12, and that seed gives both the highest CQKR score and the smallest deficit, −0.0153. The deficit remains negative in all five seeds.

Table 8. *Ridge sweep for CQKR and RBF-KR, seed 42. Validation scores only; the test split was evaluated at two points. The effective penalty is the ridge coefficient divided by the mean diagonal of $Z^\top Z$.*

Ridge	Validation, CQKR	Validation, RBF-KR	Effective penalty, CQKR	Effective penalty, RBF-KR
10^{-5}	0.5840	0.6295	2.86×10^{-6}	3.66×10^{-7}
10^{-4}	0.5840	0.6296	2.86×10^{-5}	3.66×10^{-6}
10^{-3}	0.5840	0.6303	2.86×10^{-4}	3.66×10^{-5}
10^{-2}	0.5840	0.6330	2.86×10^{-3}	3.66×10^{-4}
10^{-1}	0.5844	0.6330	2.86×10^{-2}	3.66×10^{-3}

Ridge	Validation, CQKR	Validation, RBF-KR	Effective penalty, CQKR	Effective penalty, RBF-KR
1	0.5848	0.6294	2.86×10^{-1}	3.66×10^{-2}
10	0.5749	0.6200	2.86×10^{0}	3.66×10^{-1}
10^2	0.5495	0.6067	2.86×10^{1}	3.66×10^{0}

Table 9. *Gate-0 in each seed. For CQKR: the admitted bandwidth with the largest alignment, its spread relative to 2^{-10}, its shot budget $(5/\sigma_K)^2$, the resulting test score and the net effect against RBF-KR with tuned ridge. For RUQK-NN: the alignment at $\gamma = 0.05$, the only bandwidth whose spread exceeds 0.010 in any seed, relative to the RBF reference on the same sample.*

Seed	CQKR γ^*	$\sigma_K /2^{-n}$	Shots	CQKR $\bar{F}_{\max}$	Δ, tuned	RUQK KTA / RBF at $\gamma = 0.05$	RUQK-NN
42	0.35	11.9	184,961	0.5891	−0.0466	0.92	admitted
0	0.20	83.4	3,766	0.6168	−0.0153	0.51	rejected
1	0.35	10.8	223,039	0.5823	−0.0463	0.72	rejected
2	0.35	13.7	139,348	0.5879	−0.0457	0.68	rejected
3	0.35	11.2	210,013	0.5872	−0.0471	0.89	admitted

Information bottleneck. RUQK-NN consumes all 480 raw features through 48 re-uploading layers rather than a ten-dimensional projection, so the bottleneck a low-dimensional embedding would impose is absent. In the seeds where it is admitted it still loses by 0.0603 ± 0.0005 against RBF-kNN-480 on exactly those features. Gate-0 decides how many such seeds there are, and the answer is itself a result: it admits RUQK-NN in only two of five seeds, 42 and 3, and in seeds 0, 1 and 2 no bandwidth on the grid passes the screen, so by the prespecified screening rule no comparison is made. The remaining two are not a truncated run but the protocol's outcome. The per-seed screen, reported in Table 9, shows what decides admission. In every seed the 48-layer kernel concentrates at every bandwidth from 0.10 upwards, with a spread between 1.0 and 6.6 times 2^{-10} and a shot budget of at least 596,147 per kernel entry. Only $\gamma = 0.05$ escapes concentration, with a spread 53.8 to 63.8 times the floor and roughly 6,400 to 9,300 shots, and at that bandwidth the kernel's alignment with the labels is 0.51 to 0.92 of the RBF reference. It clears the 0.80 threshold in seeds 42 and 3 and falls short in seeds 0, 1 and 2. The rejections are therefore not concentration alone: the one bandwidth at which the deep kernel is measurable is also one at which it aligns weakly with the labels, and whether it passes depends on how strong the classical reference is on that seed's 512-protein sample, which ranges from 0.1016 to 0.1770.

Dissection of the fusion gain. Only fusion produces a positive result, so we ask what it transmits. Table 10 compares the three fusion rules against Cos-kNN, and Table 11 repeats CG-QCF over five seeds against a control that is deterministic, so the paired test reduces to a one-sample test on Δ. Figure 6 shows the weight sweep, the per-seed gains and the fused system beside its members.

Table 10. *Fusion of the quantum kernel with the classical cosine kernel, against Cos-kNN at 0.6663, seed 42 with paired bootstrap intervals.*

Variant	$\bar{F}_{\max}$	Δ	95% interval	Significant
K-Mix, best $\alpha = 1.00$	0.6663	+0.0000	[+0.0000, +0.0000]	no
QCF, global $\alpha = 0.80$	0.6699	+0.0036	[+0.0022, +0.0051]	yes
CG-QCF, α per confidence quartile	0.6696	+0.0033	[+0.0016, +0.0047]	yes
CG-QCF minus QCF	—	−0.0003	[−0.0012, +0.0003]	no

Table 11. *CG-QCF over five seeds. The control is deterministic at the values shown.*

Aspect	Control	CG-QCF, mean ± s.d.	Δ	t	p (t)	p (Wilcoxon)	Cohen's d
MFO	0.7293	0.7334 ± 0.0011	+0.0041	8.38	0.0011	0.0625	3.75
BPO	0.5624	0.5663 ± 0.0008	+0.0039	10.89	0.0004	0.0625	4.87
CCO	0.7073	0.7094 ± 0.0006	+0.0021	7.33	0.0018	0.0625	3.28
Aspect mean	0.6663	0.6697 ± 0.0006	+0.0034	13.10	0.0002	0.0625	5.86

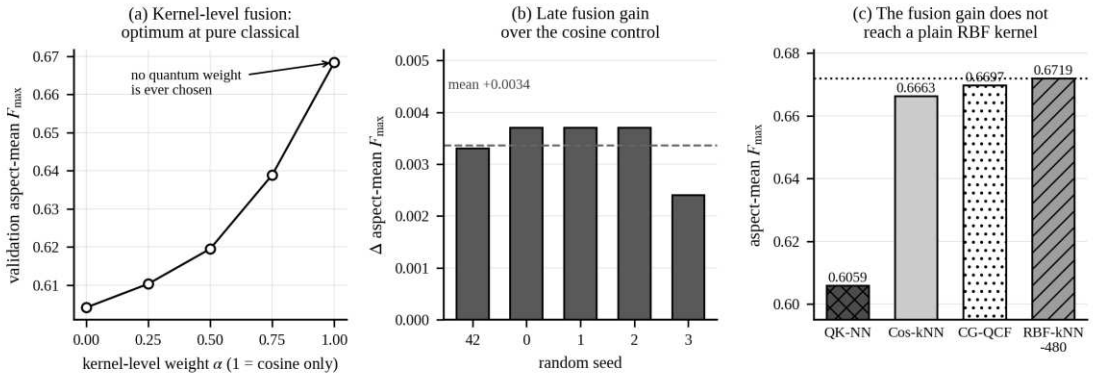


Fig 6. *(a) Validation performance of K-Mix across the classical mixing weight; the optimum sits at pure classical weight. (b) The CG-QCF gain over Cos-kNN in each of five seeds. (c) The fused system beside its members and beside RBF-kNN-480, which it does not reach.*

Three observations constrain what the gain transmits, and together they are consistent with a contribution from score-level complementarity. The first and sharpest is the contrast between the fusion rules. K-Mix and CG-QCF combine exactly the same two kernels on exactly the same data and differ only in *where* the combination happens. Mixing before prediction searches over how much quantum similarity structure to admit, and returns $\alpha = 1.00$, pure classical weight, with a net effect of $+0.0000$; mixing after two independently fitted predictors have produced probabilities gains $+0.0034$. If the quantum embedding induced a better notion of similarity between proteins, the kernel-level rule is where it would show, and it shows nothing. The second observation is the direction of the effect relative to standalone quality: QK-NN alone reaches 0.6059, far below Cos-kNN at 0.6663, yet adding a fifth to a quarter of its score improves the combination. A weaker model improving a stronger one when their scores are averaged is the defining behaviour of an ensemble and can arise when the two err on different proteins. The third bounds the gain in absolute terms: CG-QCF reaches 0.6697, while RBF-kNN-480 reaches 0.6719 in 0.2 s with no circuit and no fusion. Granting each aspect its own weights does not help either, reaching 0.6694 ± 0.0004 against 0.6697 ± 0.0006 ($p = 0.33$).

We state the limit of this reading precisely. The diversity interpretation is supported by the fusion-rule asymmetry and by the standalone ordering, but the experiment that would settle it — substituting for QK-NN a classical predictor of comparable standalone accuracy and deliberately decorrelated errors — has not been run, and K-Mix was evaluated on a five-point weight grid whereas QCF and CG-QCF used 21 points. We report diversity as the best-supported reading rather than as a demonstrated result.

4.7 Complexity analysis

For one protein the classical stage costs $\mathcal{O}(D)$ for pooling, a nearest-neighbour query costs $\mathcal{O}(N_{\text{train}}D)$, and a Nyström prediction costs $\mathcal{O}(m)$ kernel evaluations. State-vector simulation of an n-qubit circuit costs $\mathcal{O}(2^n)$ per gate, so a fidelity kernel over N_{train} neighbours costs $\mathcal{O}(N_{\text{train}}\, 2^n)$ per query in simulation. On hardware each kernel entry must be estimated from M_{shot} repetitions, which (20) ties directly to the kernel's spread. Table 12 reports Gate-0 for CQKR, and Figure 7 plots it.

Table 12. *Gate-0 for CQKR, seed 42. Concentration floor* $2^{-10} = 0.000977$*; RBF reference alignment 0.0984; shots are* $(5/\sigma_K)^2$.

γ	Mean K	σ_K	$\sigma_K/2^{-n}$	KTA	KTA / RBF	Shots required	Verdict
0.05	0.83135	0.05645	57.8	0.0750	0.76	7,844	reject
0.10	0.48142	0.13276	135.9	0.0839	0.85	1,418	admit
0.20	0.07359	0.08541	87.5	0.1056	1.07	3,427	admit
0.35	0.00425	0.01163	11.9	0.1169	1.19	184,961	selected
0.60	0.00121	0.00264	2.7	0.1152	1.17	3,575,234	reject
1.00	0.00104	0.00190	1.9	0.1141	1.16	6,934,585	reject

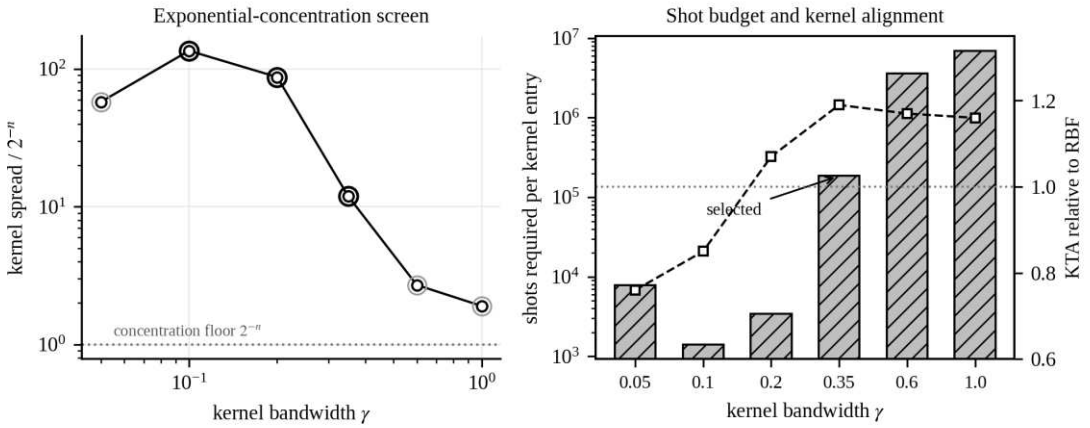


Fig 7. *Gate-0 for CQKR. Left, kernel spread relative to the 2^{-n} concentration floor across the bandwidth grid; heavy circles mark admitted configurations. Right, the shot budget $(5/\sigma_K)^2$ per kernel entry and the alignment relative to the RBF reference.*

The trade-off is structural on this task. The bandwidth that maximises alignment, $\gamma = 0.35$, sits where the kernel spread has fallen to 11.9 times the floor and a hardware execution would need 184,961 shots per kernel entry; across the five seeds of the fusion experiment the requirement ranges from 139,348 to 223,039. The bandwidths with comfortable spread, 0.10 and 0.20, need only 1,418 and 3,427 shots but align worse with the labels. The five-seed screen of Table 9 adds a caution about the selection rule itself. Maximising alignment chooses $\gamma = 0.35$ in four seeds, at 139,348 to 223,039 shots, but in seed 0 the two bandwidths align almost equally and the rule falls on $\gamma = 0.20$, at 3,766 shots. That seed produces the best CQKR score of the five. The finite-shot simulation below adds a second line of evidence in seed 42, where forcing $\gamma = 0.20$ raises QK-NN from 0.6059 to 0.6160 and CQKR from 0.5891 to 0.6178. Both observations point the same way as the concentration literature (Thanasilp et al. 2024; Gil-Fuster et al. 2024): alignment measured on 512 training proteins is a noisy criterion, and it tends to select the more concentrated and less measurable kernel, which here also generalised worse. We report this as an observation on two seeds rather than an established result. Table 13 converts the measured wall-clock into cost against each control, and Figure 8 plots accuracy against it.

Table 13. *Measured fitting cost of each quantum model against its matched control. Times are from one run; Δ is the five-seed mean from Table 7 (two seeds for RUQK-NN; QRes against PM-MLP computed from the same seed-wise values).*

Pair	Quantum (s)	Control (s)	Ratio	Δ
VQNN vs PM-MLP	1,345.8	17.3	78×	−0.0023
QRes vs PM-MLP	1,071.5	17.3	62×	−0.0015
CQKR vs RBF-KR	0.6	0.1	6×	−0.0370
QK-NN vs RBF-kNN-10	0.4	0.1	4×	−0.0186
RUQK-NN vs RBF-kNN-480	9.6	0.2	48×	−0.0603
CG-QCF vs Cos-kNN	10.4	0.5	21×	+0.0034

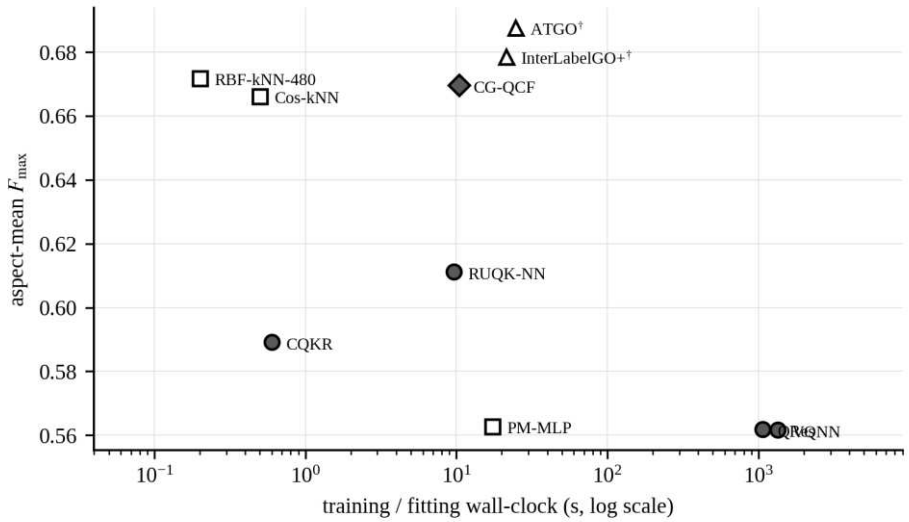


Fig 8. *Accuracy against fitting wall-clock on a logarithmic scale. No model containing a quantum component occupies the upper-left region that RBF-kNN-480 and Cos-kNN share.*

Finite-shot estimation. The shot budgets above are derived per kernel entry; we also measured what finite shots do to the models themselves. On hardware each fidelity-kernel entry is estimated from M runs of a compute–uncompute circuit whose all-zeros outcome has probability K_{ij}, so we replace every entry of every quantum kernel — validation, test and landmark — by

$$\widehat{K}_{ij} = \frac{1}{M}\,\mathrm{Binomial}\big(M,\, K_{ij}\big), \tag{25}$$

draw each entry independently, and rerun every downstream step unchanged, including the selection of the fusion weights on the noisy validation kernel. This models shot noise only, without gate error, decoherence or readout error, so it bounds the hardware penalty from below. We sweep M from 10^2 to 10^6 with three independent noise draws per value, at the bandwidth Gate-0 selected in seed 42 and at the more measurable $\gamma = 0.20$; the classical controls are noise-free. Without noise the simulation reproduces every quantum and classical figure of seed 42 exactly. Table 14 and Figure 9 report the result.

Table 14. *Finite-shot simulation, seed 42. Mean over three noise draws; Δ against the noise-free controls RBF-kNN-10 (0.6262), Cos-kNN (0.6663) and RBF-KR (0.6313). CQKR shows mean ± s.d. over draws.*

γ	M	QK-NN	Δ	CG-QCF	Δ	CQKR	Δ
0.35	10^2	0.6040	−0.0221	0.6694	+0.0031	0.5792 ± 0.0009	−0.0521
0.35	10^3	0.6057	−0.0205	0.6698	+0.0034	0.5878 ± 0.0005	−0.0434
0.35	10^4	0.6055	−0.0206	0.6697	+0.0033	0.5891 ± 0.0003	−0.0422
0.35	10^6	0.6058	−0.0203	0.6697	+0.0034	0.5891 ± 0.0001	−0.0422
0.35	∞	0.6059	−0.0203	0.6696	+0.0033	0.5891	−0.0422
0.20	10^2	0.6148	−0.0114	0.6699	+0.0036	0.2956 ± 0.0313	−0.3357
0.20	10^3	0.6157	−0.0105	0.6697	+0.0034	0.2981 ± 0.0426	−0.3332
0.20	10^4	0.6159	−0.0103	0.6696	+0.0033	0.3877 ± 0.0583	−0.2436
0.20	10^5	0.6160	−0.0101	0.6695	+0.0032	0.6173 ± 0.0002	−0.0139
0.20	∞	0.6160	−0.0102	0.6697	+0.0033	0.6178	−0.0135

Three results follow. First, the nearest-neighbour models barely notice shot noise: QK-NN loses at most 0.0019 at 100 shots per entry and is within 0.0002 of its noise-free value from 10^3 onwards, because a k-nearest-neighbour vote depends only on which training proteins rank highest, and sampling error rarely reorders them. The late-fusion gain therefore survives finite shots, at $+0.0031$ with 100 shots per entry against $+0.0033$ without noise, which is consistent with the diversity reading of Section 4.6: decorrelated errors do not require precise similarities. Second, the Nyström regressor is fragile in a way the per-entry budget does not predict. At $\gamma = 0.35$ the landmark kernel is close to the identity, its inverse square root is well conditioned, and CQKR recovers fully by 10^4 shots, below its 5σ budget of 184,961. At $\gamma = 0.20$ the landmark kernel carries real off-diagonal structure, small eigenvalues appear, and $K_{mm}^{-1/2}$

amplifies sampling error: CQKR collapses to about 0.30 up to 10^3 shots and recovers only at 10^5, roughly thirty times its 5σ budget of 3,427. The per-entry budget of Gate-0 is therefore neither an upper nor a lower bound on what a downstream estimator needs. Third, the ordering against the classical controls holds at every M: no quantum model reaches its control at any shot count, and the one gain, that of fusion, is present at every shot count because it never depended on accurate quantum similarities. Even at the 100 shots the nearest-neighbour models tolerate, a single set of test predictions requires 3.36×10^8 kernel entries against the training set, or 3.4×10^{10} circuit executions.

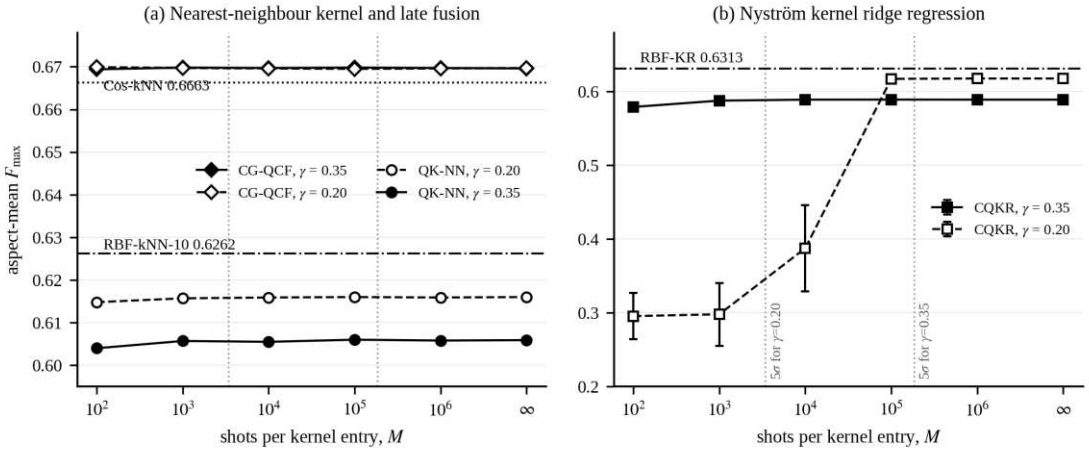


Fig 9. *Finite-shot simulation in seed 42, mean over three noise draws. (a) QK-NN and CG-QCF are nearly insensitive to the shot count at both bandwidths; the fusion gain over Cos-kNN is present at every M, and QK-NN stays below RBF-kNN-10. (b) CQKR at $\gamma = 0.20$ collapses until about 10^5 shots, well beyond its per-entry 5σ budget, while at $\gamma = 0.35$ it recovers by 10^4. Dotted vertical lines mark the Gate-0 budgets.*

4.8 Visualization analysis

Figure 4 establishes the ranking: every standalone quantum model sits below Cos-kNN, and the only model containing a quantum component that approaches the classical nearest-neighbour baselines is the fused CG-QCF, which still sits below RBF-kNN-480. Figure 5 converts the ranking into net effects over seeds and makes the matched comparisons visible directly; every quantum-against-classical interval lies entirely left of zero, and the one interval that contains zero belongs to the trainability ablation, which is expected to be null. Figure 6 shows why the positive result is not evidence of quantum similarity: panel (a) rises monotonically to pure classical weight, panel (b) shows that the late-fusion gain repeats in every seed, and panel (c) places the fused system below a plain RBF kernel. Figure 7 shows that the alignment optimum and the measurability optimum sit at different bandwidths, which is the structural reason no configuration offers both. Figure 8 shows the cost: the quantum models occupy the right half of the plot, up to three orders of magnitude slower, without gaining accuracy. We deliberately do not report a receiver-operating-characteristic curve or a confusion matrix; under a label density near 0.10 across 180 terms, true negatives dominate the flattened matrix and inflate ROC area, and a confusion matrix is ill-defined for a problem in which each protein carries roughly 17 to 19 simultaneous positive terms.

5. Discussion and Conclusion

The pattern across five quantum models is consistent enough to carry more weight than any single comparison. Each model was chosen because it addresses a different suspected cause of failure — the variational model tests whether a trainable circuit helps, the covariant kernel tests whether group structure helps, the re-uploading kernel removes the information bottleneck of a low-dimensional projection, and the feature-map arm tests whether learned quantum features transfer to a different classifier. Each lost to its own control in every admitted seed, and the losses are not marginal: the largest, -0.0847 ± 0.0056 for learned quantum features against a classical surrogate of identical width, exceeds the 0.0828 that separates logistic regression from a random forest. The tested configurations do not show a standalone quantum advantage on this internal split; the results do not exclude a gain from other circuits, representations or data partitions.

Three results close off the explanations a reader would reach for first. QRes is indistinguishable from the trained VQNN over five seeds, $+0.0008 \pm 0.0016$, although the trained circuit moved by 9.295×10^{-1} against 1.863×10^{-9} for the frozen one, which argues against a simple failure of the parameters to move during optimisation. RUQK-NN consumes all 480 raw features and still loses by 0.0603 ± 0.0005 to RBF-kNN-480 on exactly those features in the seeds Gate-0 admits, which weakens the low-dimensional bottleneck explanation in the admitted seeds; in the other three seeds the only bandwidth at which the 48-layer kernel escapes concentration aligns too weakly with the labels to be admitted. And a separately tuned regulariser widens the CQKR deficit in every seed, from -0.0370 to

-0.0402 on average, which argues against the hypothesis that a single ridge coefficient silently handicapped the quantum kernel despite the two kernels' hundredfold difference in scale. What remains is the kernel itself: on this data, the similarity structure a shallow quantum embedding induces is less useful than the one a tuned RBF or cosine kernel induces. Gate-0 explains why quantitatively rather than by appeal: the bandwidth that best aligns the quantum kernel with the labels already sits close to the 2^{-n} floor and would require roughly 1.8×10^5 shots per kernel entry, while bandwidths with comfortable measurability align worse than a classical RBF reference, exactly as the concentration literature predicts (Thanasilp et al. 2024; Gil-Fuster et al. 2024). The one seed in which the rule selected the less concentrated bandwidth also produced the best quantum kernel of the five, and forcing that bandwidth in seed 42 improves both kernel models. A selection rule weighting measurability as well as alignment would therefore serve the quantum models better, but it would not, on this evidence, close the gap to the classical controls: at $\gamma = 0.20$ QK-NN still trails RBF-kNN-10 by 0.0102 and CQKR trails RBF-KR by 0.0135. Measurability also carries a cost of its own. The less concentrated landmark kernel makes the Nyström inversion ill-conditioned, so CQKR at $\gamma = 0.20$ needs about 10^5 shots per entry before it recovers — thirty times the per-entry budget Gate-0 reports.

The one positive result is the most instructive part of the study, because it is consistent across the five tested seeds even though its mechanism remains unresolved. Late fusion gains +0.0034 aspect-mean $F_{\max}$ across five seeds with an effect size of 5.86 and $p = 0.0002$, and reported alone that line would read as a quantum advantage. It does not establish a quantum advantage: the same two kernels combined before prediction gain exactly zero at pure classical weight, the gain appears only once two independently fitted predictors have their scores averaged, and the fused system at 0.6697 is still beaten by RBF-kNN-480 at 0.6719. We draw a methodological lesson from this rather than only an empirical one. A significance test establishes that an effect repeats, not that it arises from the component a paper credits; the discriminating evidence here came not from more seeds or a tighter interval but from a second experiment — kernel-level fusion — built so that the two candidate mechanisms would give different answers. Wherever a quantum component is one part of a larger pipeline, we would pair every fusion result with the corresponding pre-fusion combination, since the pair separates a contribution to the representation from a contribution to the ensemble. For practitioners the ordering of investment on this task is unambiguous: Cos-kNN reaches 0.6663 in 0.5 s, RBF-kNN-480 reaches 0.6719 in 0.2 s, and no quantum model reaches either. For the applied quantum machine learning community, the matching rule and Gate-0 cost little to add and change conclusions substantially.

Several constraints bound what the present study establishes. Every comparison rests on five seeds and one split, with the exception of RUQK-NN, which Gate-0 admitted in two seeds; five seeds cannot reach $p < 0.05$ under a two-sided Wilcoxon test, so the non-parametric result does not independently establish significance, and the variation we report is over initialisations and seed-dependent choices rather than over independent data splits. K-Mix was swept on a five-point weight grid while the late-fusion rules used 21 points, so the kernel-level result is established at a coarser resolution than the result it is contrasted with. We evaluate 180 frequent terms with the roots excluded on an internally constructed split, so our figures do not compare with official CAFA results, and rare-term behaviour — the regime in which quantum methods have most often been argued to help — remains untested; the partition is cluster-disjoint and audited, but its clustering does not certify a sequence-identity threshold and it is a retrospective repartition rather than a prospective time split. The reimplementations of ATGO and InterLabelGO+ omit components their authors consider load-bearing, so they under-represent the published systems. All quantum models run on a state-vector simulator with analytic expectation values, without device noise, gate error or readout error, and our finite-shot simulation captures sampling error only; hardware errors could further change performance. Finally, our conclusions are bounded by the three feature-map families we tested at ten qubits; a substantially deeper circuit or a problem-specific ansatz could behave differently, and nothing here rules that out.

We rank the resulting research directions by expected value. Settling the mechanism behind the fusion gain comes first, since it is the one interpretive claim this paper leaves open: measure the error correlation between the quantum and classical predictors directly, substitute for QK-NN a classical predictor of comparable standalone accuracy and deliberately decorrelated errors, and rerun K-Mix on the same 21-point grid; if the gain survives the substitution, it is not quantum in origin. Adding gate, decoherence and readout noise to the finite-shot simulation comes second, since shot noise alone leaves the nearest-neighbour models almost untouched and hardware error is the remaining unknown; the same runs can test, across seeds, whether a Gate-0 rule that prefers measurable bandwidths among near-equal alignments improves the quantum kernels. Repeating the benchmark over independent cluster-disjoint splits comes third, so that the variance we report extends from initialisations to data, and extending the label set towards rare and deep terms comes fourth.

On this internal hierarchical GO benchmark, the tested quantum components do not improve on their corresponding classical controls. Four model comparisons use five seeds; the 48-layer re-uploading model passes the prespecified kernel screen in two of five seeds and loses to its control in both. Differences across the matched pairs range from 0.0023 to 0.0847 in aspect-mean maximum F-measure, with the largest deficit for quantum features against an equal-width classical surrogate. Late fusion of quantum and classical scores improves on cosine nearest neighbours by 0.0034 across five seeds, yet the fused score remains below that of a classical RBF nearest-neighbour model. The fusion-rule comparison is consistent with an ensemble explanation, but a matched classical substitute is needed to establish its

mechanism. Gate-0 exposes a trade-off between label alignment and kernel measurability, while finite-shot simulation shows that the downstream Nyström model may require substantially more shots than the kernel-spread heuristic suggests. These findings extend our earlier single-feature-map preprint (Majidi 2026) through multiple model families, matched controls and an explicit resource analysis. They apply to one retrospective split and simulated circuits; prospective CAFA evaluation and hardware noise remain open.

Statements and Declarations

-Funding

The author declares that no funds, grants, or other support were received during the preparation of this manuscript.

-Competing Interests

The author has no relevant financial or non-financial interests to disclose.

-Data Availability

The CAFA6 data used in this study are publicly available from the challenge organisers. The derived artefacts required to reproduce every number in Tables 3 and 6–14 — the split manifests with their SHA-256 input digests, the cluster assignments and split audit, the 180 selected GO terms with their aspect assignments, the information-content table and the per-seed result records — together with the code supporting the findings of this study are available from the corresponding author upon reasonable request.

References

S. F. Altschul et al. (1997) Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. Nucleic Acids Research, 25, 17, 3389–3402.

E. R. Anschuetz and B. T. Kiani (2022) Quantum variational algorithms are swamped with traps. Nature Communications, 13, 1, article 7760.

M. Ashburner et al. (2000) Gene ontology: Tool for the unification of biology. Nature Genetics, 25, 1, 25–29, https://doi.org/10.1038/75556.

V. Bergholm et al. (2018) PennyLane: Automatic differentiation of hybrid quantum-classical computations. arXiv:1811.04968.

J. Bowles, S. Ahmed, and M. Schuld (2024) Better than classical? The subtle art of benchmarking quantum machine learning models. arXiv:2403.07059.

A. Z. Broder (1997) On the resemblance and containment of documents. in Proc. Compression and Complexity of Sequences, IEEE, 21–29.

R. Cao et al. (2017) ProLanGO: Protein function prediction using neural machine translation based on a recurrent neural network. Molecules, 22, 10, article 1732.

Y. Cao and Y. Shen (2021) TALE: Transformer-based protein function annotation with joint sequence–label embedding. Bioinformatics, 37, 18, 2825–2833.

M. Cerezo, G. Verdon, H.-Y. Huang, L. Cincio, and P. J. Coles (2022) Challenges and opportunities in quantum machine learning. Nature Computational Science, 2, 567–576.

M. Cerezo et al. (2025) Does provable absence of barren plateaus imply classical simulability?. Nature Communications, 16, article 7907.

A. Chervov, A. Vakhrushev, S. Fironov, and L. Martignetti (2024) ProtBoost: Protein function prediction with Py-Boost and graph neural networks — CAFA5 top 2 solution. arXiv:2412.04529, https://doi.org/10.48550/arXiv.2412.04529.

W. T. Clark and P. Radivojac (2013) Information-theoretic evaluation of predicted ontological annotations. Bioinformatics, 29, 13, i53–i61.

M. C. De Paolis Kaluza et al. (2026) Advances in protein function prediction from the fifth CAFA challenge. bioRxiv, https://doi.org/10.64898/.04.27.716980.

H. Doga et al. (2024) A perspective on protein structure prediction using quantum computers. Journal of Chemical Theory and Computation, 20, 9, 3359–3378.

E. Fontana et al. (2024) Characterizing barren plateaus in quantum ansätze with the adjoint representation. Nature Communications, 15, 1, article 7171.

E. Gil-Fuster, J. Eisert, and V. Dunjko (2024) On the expressivity of embedding quantum kernels. Machine Learning: Science and Technology, 5, article 025003.

J. R. Glick et al. (2024) Covariant quantum kernels for data with group structure. Nature Physics, 20, 479–483.

V. Gligorijević et al. (2021) Structure-based protein function prediction using graph convolutional networks. Nature Communications, 12, 1, article 3168.

V. Havlíček et al. (2019) Supervised learning with quantum-enhanced feature spaces. Nature, 567, 7747, 209–212.

Z. Holmes, K. Sharma, M. Cerezo, and P. J. Coles (2022) Connecting ansatz expressibility to gradient magnitudes and barren plateaus. PRX Quantum, 3, 1, article 010313.

H.-Y. Huang et al. (2021) Power of data in quantum machine learning. Nature Communications, 12, article 2631, https://doi.org/10.1038/s41467-021-22539-9.

T. Hubregtsen, D. Wierichs, E. Gil-Fuster, P.-J. H. Derks, P. K. Faehrmann, and J. J. Meyer (2022) Training quantum embedding kernels on near-term quantum computers. Physical Review A, 106, article 042431.

S. Jerbi, L. J. Fiderer, H. Poulsen Nautrup, J. M. Kübler, H. J. Briegel, and V. Dunjko (2023) Quantum machine learning beyond kernel methods. Nature Communications, 14, article 517, https://doi.org/10.1038/s41467-023-36159-y.

Y. Jiang et al. (2016) An expanded evaluation of protein function prediction methods shows an improvement in accuracy. Genome Biology, 17, 1, article 184.

J. Jäger and R. V. Krems (2023) Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines. Nature Communications, 14, article 576.

M. Kulmanov and R. Hoehndorf (2020) DeepGOPlus: Improved protein function prediction from sequence. Bioinformatics, 36, 2, 422–429.

M. Kulmanov and R. Hoehndorf (2022) DeepGOZero: Improving protein function prediction from sequence and zero-shot learning based on ontology axioms. Bioinformatics, 38, supp. 1, i238–i245.

M. Kulmanov et al. (2024a) Predicting protein functions using positive-unlabelled ranking with ontology-based priors. Bioinformatics, 40, supp. 1, i401–i409.

M. Kulmanov et al. (2024b) Protein function prediction as approximate semantic entailment. Nature Machine Intelligence, 6, 220–228.

M. Larocca et al. (2025) Barren plateaus in variational quantum computing. Nature Reviews Physics, 7, 4, 174–189, https://doi.org/10.1038/s42254-025-00813-9.

Z. Lin et al. (2023) Evolutionary-scale prediction of atomic-level protein structure with a language model. Science, 379, 6637, 1123–1130.

Y.-H. Liu et al. (2020) GODoc: High-throughput protein function prediction using novel k-nearest-neighbor and voting algorithms. BMC Bioinformatics, 21, supp. 6, article 276.

Q. Liu, C. Zhang, and L. Freddolino (2024) InterLabelGO+: Unraveling label correlations in protein function prediction. Bioinformatics, 40, 11, article btae655, https://doi.org/10.1093/bioinformatics/btae655.

Majidi F (2026) A Dimension-Matched Quantum Feature Map for Hierarchical Gene Ontology Term Prediction: Attributing the Gain on CAFA6. Research Square preprint. https://doi.org/10.21203/rs.3.rs-10501636/v1

J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven (2018) Barren plateaus in quantum neural network training landscapes. Nature Communications, 9, article 4812.

P. Radivojac et al. (2013) A large-scale evaluation of computational protein function prediction. Nature Methods, 10, 3, 221–227.

M. Ragone et al. (2024) A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits. Nature Communications, 15, 1, article 7172.

A. Rahimi and B. Recht (2007) Random features for large-scale kernel machines. in Advances in Neural Information Processing Systems, 20, 1177–1184.

R. Ramola et al. (2026) On the state of protein function prediction: A report on the fourth CAFA challenge. bioRxiv.05.06.722942.

A. Rives et al. (2021) Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proceedings of the National Academy of Sciences, 118, 15, article e2016239118.

A. Robert et al. (2021) Resource-efficient quantum algorithm for protein folding. npj Quantum Information, 7, 1, article 38.

F. J. Schreiber, J. Eisert, and J. J. Meyer (2023) Classical surrogates for quantum learning models. Physical Review Letters, 131, 10, article 100803.

M. Schuld and N. Killoran (2019) Quantum machine learning in feature Hilbert spaces. Physical Review Letters, 122, 4, article 040504.

R. Shaydulin and S. M. Wild (2022) Importance of kernel bandwidth in quantum machine learning. Physical Review A, 106, article 042407.

P. Singkanipa and D. A. Lidar (2025) Beyond unital noise in variational quantum algorithms: Noise-induced barren plateaus and limit sets. Quantum, 9, article 1617.

S. Thanasilp, S. Wang, M. Cerezo, and Z. Holmes (2024) Exponential concentration in quantum kernel methods. Nature Communications, 15, article 5200.

M. van Kempen et al. (2024) Fast and accurate protein structure search with Foldseek. Nature Biotechnology, 42, 2, 243–246.

L. Wei et al. (2023) Quantum machine learning in medical image analysis: A survey. Neurocomputing, 525, 42–53, https://doi.org/10.1016/j.neucom..01.049.

H. Yan, S. Wang, H. Liu, H. Mamitsuka, and S. Zhu (2024) GORetriever: Reranking protein-description-based GO candidates by literature-driven deep information retrieval for protein function annotation. Bioinformatics, 40, supp. 2, ii53–ii61, https://doi.org/10.1093/bioinformatics/btae401.

Q. Yuan et al. (2023) Fast and accurate protein function prediction from sequence through pretrained language model and homology-based label diffusion. Briefings in Bioinformatics, 24, 3, article bbad117.

Y. Zhang et al. (2025) A hybrid quantum-AI framework for protein structure prediction on NISQ devices. arXiv:2510.06413.

Y. Zhang et al. (2026) A quantum framework for protein binding-site structure prediction on utility-level quantum processors. Advanced Science, 13, 12, article e13641, https://doi.org/10.1002/advs.202513641.

N. Zhou et al. (2019) The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. Genome Biology, 20, 1, article 244, https://doi.org/10.1186/s13059-019-1835-8.

Y.-H. Zhu et al. (2022) Integrating unsupervised language model with triplet neural networks for protein gene ontology prediction. PLOS Computational Biology, 18, 12, article e1010793.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Algorithm2.svg
- Algorithm1.svg
- Algorithm4.svg
- Algorithm5.svg
- Algorithm3.svg
- quantumcafa.ipynb