

EIHR-PROT: Explicitly Modelling of Homology Reliability for Protein Function Prediction

Oluranti Jonathan

Covenant University

Uwidia . E. Osalodion

`osalodion.uwidiapgs@stu.cu.edu.ng`

Covenant University

Research Article

Keywords: protein function prediction, gene ontology, protein annotation, ESM-2

Posted Date: August 21st, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-10341355/v1>

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

[Read Full License](#)

Additional Declarations: No competing interests reported.

1 **EIHR-PROT: Explicitly Modelling of Homology Reliability for Protein Function**

2 **Prediction**

3 ¹Oluranti Jonathan, ²Uwidia .E. Osalodion

4 ^{1,2}Department of Computer and Information Sciences, Covenant University, Ota, Ogun State,
5 Nigeria

6 ¹jonathan.oluranti@covenantuniversity.edu.ng

7 ²osalodion.uwidiapgs@stu.cu.edu.ng

8 Corresponding Author: Uwidia .E. Osalodion

9 Corresponding Author Email: osalodion.uwidiapgs@stu.cu.edu.ng

10 **Abstract**

11 **Background:** Protein function prediction remains a fundamental challenge in computational
12 biology, with deep learning methods increasingly complementing traditional homology-based
13 approaches. While protein language models (PLMs) such as ESM-2 have demonstrated
14 sufficient understanding of evolutionary patterns to be competitive for downstream functional
15 annotation tasks, homology-based inference continues to provide valuable information when
16 reliable templates exist. However, existing hybrid approaches typically employ fixed fusion
17 strategies that fail to account for the varying reliability of homology evidence across proteins.

18 We introduce EIHR-PROT (Explicitly Incorporated Homology Reliability Signals for
19 PROTein Function Prediction), a novel framework that adaptively integrates sequence-derived
20 neural predictions with homology-based priors through a learned reliability-gating mechanism.
21 Our model extracts frozen embeddings from ESM-2, processes them through an attention-
22 pooling module and neural prediction branch, while simultaneously constructing homology
23 priors from DIAMOND-based sequence similarity searches.

Results: Compared with similar methods which employ aspect-specific weights, EIHR-PROT uses explicit alignment-derived features, including bitscore, query coverage, and hit count, to dynamically weight each modality's contribution on a per-protein basis.

Evaluation on the PDB dataset demonstrates that EIHR-PROT outperforms existing sequence and homology-based methods, DeepGOPlus and TALE+, across all Gene Ontology (GO) aspects with Fmax scores of 0.604, 0.676, 0.731, for BP, MF, and CC, respectively. Ablation studies confirm that explicit reliability signals improve performance, particularly for Biological Process (BP) prediction, while gate weight analysis reveals appropriate down-weighting of homology contributions in low-similarity regimes.

Conclusion: Beyond improving predictive performance, our findings demonstrate that explicitly modelling homology reliability provides a principled and interpretable framework for adaptive evidence integration in hybrid protein function prediction. Implementation of our model can be found at <https://github.com/uwida/EIHR-PROT>.

Keywords: protein function prediction, gene ontology, protein annotation, ESM-2

Background

Machine learning-based methods have substantially reshaped automated protein function prediction, particularly for Gene Ontology (GO) annotation. Sequence-based protein function prediction, in particular, has become increasingly attractive due to the development of protein language models (PLMs) such as ESM-2 and ProtT5.

These models, which are pretrained on large-scale unlabeled protein corpora, learn biologically meaningful representations directly from amino acid sequences [1, 2]. Embeddings from PLMs implicitly capture patterns associated with evolutionary conservation, secondary structure,

residue contacts, localization signals, and other functional determinants that are strongly linked to protein activity [3,4].

Consequently, downstream architectures built on frozen or lightly fine-tuned protein sequence representations can achieve state-of-the-art GO prediction performance even with simple downstream classifiers [5]. Several protein function prediction methods combine sequence-embeddings with additional modalities, such as structure, protein-protein interaction networks, GO hierarchy information, textual definitions, and domain annotations [6-9].

Although these multimodal architectures have reported further gains in several contexts, the magnitude of improvement over strong PLM baselines is often modest and highly context-dependent [10]. Structural information has shown particular value for Molecular Function prediction and spatially dependent tasks, while protein-protein interaction (PPI)-derived information often contributes more strongly to Biological Process prediction [11,12]. In Critical Assessment of Protein Function Annotation (CAFA)-style evaluations, such methods often achieve the highest absolute performance, although well-implemented homology-based and unimodal protein language model approaches can still approach or match multimodal performance in some settings [13].

However, multimodal systems typically require more complex architectural designs, additional biological data sources, and substantially greater computational resources than sequence-only approaches. Due to these constraints, PLM-based sequence models therefore remain an attractive balance between predictive performance and computational efficiency in many practical annotation settings [14].

Compared with methods which rely strictly on sequence similarity for functional annotation, PLM representations offer improved performance in low-homology settings where traditional

alignment-based annotation transfer becomes unreliable [15,16]. Littmann et al. [3] developed an embedding-based approach which retained useful performance below 20% sequence identity, suggesting that learned protein representations can capture functional signals not available through direct sequence alignment alone.

Multiple comparative studies have also shown that PLM-based methods substantially outperform BLAST and related homology-transfer approaches when sequence identity between query proteins and annotated templates falls below approximately 20–30% [17]. For example, GAT-GO achieved an MF Fmax of 0.508 compared to 0.117 for BLAST on proteins sharing less than 15% sequence identity with the training data [6].

Despite the success of PLM-based methods, homology-based annotation remains central to practical protein function prediction workflows. Alignment-based tools such as BLAST and DIAMOND remain highly effective when close annotated homologs are available, particularly for Molecular Function prediction where conserved sequence motifs and catalytic residues are strongly associated with biochemical activity [13, 18].

Rather than supporting a simple replacement of homology by deep learning, the literature suggests that the optimal method depends on several factors, including sequence-similarity context, GO domain, target protein population, and the availability of reliable templates [19].

Therefore, homology-based inference and deep learning can be viewed as complementary rather than competing paradigms. Homology transfer provides highly informative evidence when reliable templates exist, whereas PLM-based models can generalize beyond direct sequence similarity by learning distributed functional representations from large-scale protein corpora [17,20].

Consistent with this complementarity, several high-performing GO predictors combine deep learning with homology-derived information. These methods provide a reliable and lightweight solution for functional annotation. Representative examples include DeepGOPlus, ATGO+, TALE+, TEMPROT+, SPROF-GO, and StarFunc, which integrate homology using approaches such as weighted ensemble prediction, late fusion, label diffusion, or consensus scoring [15, 17, 21-23].

The quantitative benefit of incorporating homology into deep learning models has been repeatedly demonstrated across GO sub-ontologies [24]. Across comparative studies, hybrid approaches typically improve Fmax or AUPR by approximately 2–15% relative to deep learning alone. However, the magnitude of improvement varies across datasets and ontologies. For example, DeepGOPlus achieves Fmax scores of 0.469, 0.544, and 0.623 compared with DeepGOCNN (0.378, 0.420, and 0.607) for biological process (BP), molecular function (MF), and cellular component (CC) ontology aspects, respectively.

Despite these advances, it is important to note that most existing hybrid architectures that combine homology-derived predictions and neural predictions do so using fixed or weakly adaptive integration strategies. Common approaches include weighted averaging, convex combinations, ensemble voting, label diffusion, or other late-fusion schemes [21,22].

To the best of our knowledge, none of the existing GO prediction hybrids have incorporated explicit alignment-derived features with a gating mechanism to condition fusion of homology and sequence modalities on a per-protein basis. This presents a limitation because, homology evidence is not equally reliable across proteins. Strong, high-confidence homologous matches provide substantially different evidence from weak, distant, or sparsely aligned hits [16]. Therefore, a fixed fusion strategy may not optimally balance neural and homology-derived predictions across varying sequence-similarity regimes.

In this article, we propose EIHR-PROT (Explicitly Incorporated Homology Reliability Signals for PROTEin Function Prediction), a sequence-homology framework for multilabel GO prediction. The model incorporates a sequence branch and a homology branch which generates GO priors from sequence-similarity evidence curated from a pre-built DIAMOND database. A learned reliability gate then receives homology reliability signals, including alignment-derived similarity metrics, and dynamically assigns instance-specific weights to the sequence-based neural prediction and the homology prior before fusion to produce a final prediction.

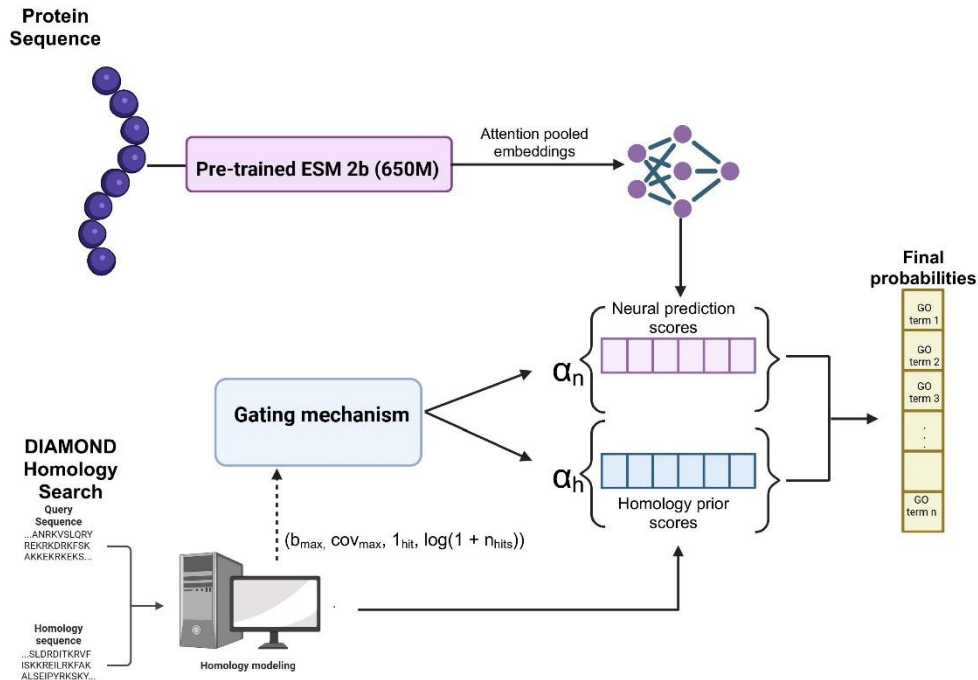


Figure 1. Architecture of EIHR-PROT. The gating mechanism produces modality-level weights for the neural prediction and the homology prior. These weights are used to compute a late-fusion probability vector, which serves as the final multilabel GO prediction for the protein. This fused prediction is then used for loss computation during training and for threshold-based evaluation during validation.

Since prior studies have already demonstrated the value of hybrid sequence–homology approaches, this study seeks to test the hypothesis that explicitly conditioning the fusion process on homology reliability signals through a learned gating mechanism will yield improved performance across all GO aspects compared with previously applied fusion methods.

We hypothesize that infusing the reliability-conditioned gating mechanism will allow EIHR-PROT to better define per-protein modality weights across different GO aspects based on each instance-specific similarity metrics and potentially provide improved balance between direct annotation transfer and learned sequence representations, especially across heterogeneous sequence-similarity regimes.

Methods

Overview of Model Architecture

The core architecture of our model included a sequence-only layer using frozen embeddings from the ESM2b (650M) protein language model. These embeddings were then pooled using the attention mechanism before prediction was made with a two-layer prediction head. We also obtained predictions from a homology branch built using a DIAMOND database constructed from the training set proteins. Both predictions were adaptively weighted via a gating mechanism and the final GO term probabilities were produced. A detailed description of the model’s architecture is seen in Figure 1.

Dataset

We used the PDB dataset from DeepFRI [7], available at the DeepFRI github repository [25]. This dataset consists of 36,640 proteins obtained from the Protein Data Bank (PDB) database. The proteins were split into training (29,902), validation (3,322), and test (3,416) sets. The

PDB dataset represents a non-redundant set. PDB chains were clustered at 95% sequence identity and a representative chain with at least one GO term from any of the GO ontologies (BP, MF, or CC) was selected from each cluster. GO terms for each protein were organized hierarchically into BP, MF, and CC ontologies. Furthermore, GO terms were filtered based on their frequency in the dataset. Only terms valid for over 50 of the PDB training dataset were selected. Overall, a total of 1973, 328, and 504 unique terms were represented across BP, CC, and MF aspects respectively. The DeepFRI PDB dataset used transferred annotations from SIFTS (structure integration with function, taxonomy, and sequence) and UniProtKB.

Test set construction

The PDB test set consisted of 3,416 PDB chains having only experimentally derived GO annotations with evidence codes: EXP (inferred from experiment), IDA (inferred from direct assay), IPI (inferred from physical interaction), IMP (inferred from mutant phenotype), IGI (inferred from genetic interaction), IEP (inferred from expression pattern), TAS (traceable author statement), and IC (inferred by curator).

The complete test set included proteins with varying levels of sequence-similarity to the training set. To determine model performance across varying levels of prior homology information, the test set proteins were organized into bins (< 30%, < 40%, < 50%, < 70%, and < 95%) based on their sequence similarity to the training set. The number of proteins across each bin is provided in Supplementary Table S1.

Protein Sequence Feature Extraction

We obtained protein sequence representations using pre-computed embeddings from ESM-2 [26]. ESM-2 is a protein language model trained on approximately 65 million proteins sampled from about 43 million Uniref50 clusters. The model provides robust vector representations that capture underlying biology, including low-resolution secondary and tertiary structure which

are valuable for downstream function prediction. For our research, this pre-trained protein language model acted as a frozen feature extractor and was not fine-tuned during training.

Scalar Attention Pooling

Protein sequences were represented using contextual residue embeddings extracted from the ESM-2 model variant with 650 million parameters (esm2_t33_650M_UR50D). For a protein of length L , the model produces a residue embedding matrix

$$\mathbf{H}^{(seq)} \in \mathbb{R}^{L \times 1280} \quad (1)$$

The resulting residue embedding matrix was passed through a trainable scalar attention-pooling module, which assigned a learned weight to each non-padded residue and computed a weighted sum to obtain a fixed-length sequence representation. The ESM-2 model architecture includes a 1024-token limit, including special BOS and EOS tokens. Therefore, our model truncates sequences longer than 1022 amino acid residues and logs the length of the truncated sequence.

The sequence branch was designed to produce a sequence-derived global representation rather than preserve residue-level outputs all the way to classification. To obtain this compact representation, the residue embedding matrix was passed through an attention-pooling module that learns how strongly each residue should contribute to the final protein-level summary.

$$\mathbf{a} = \text{softmax}\left(\text{MLP}\left(\mathbf{H}^{(seq)}\right)\right) \quad (2)$$

where $\mathbf{a} \in \mathbb{R}^L$ denotes a vector of learned weights corresponding to each protein sequence residue. The pooled sequence representation is then:

$$\mathbf{r}^{(s)} = \sum_{i=1}^L a_i \mathbf{h}_i^{(seq)} \quad (3)$$

where $\mathbf{h}_i^{(seq)}$ represents the embedding vector for each residue i in $\mathbf{H}^{(seq)}$

Neural Prediction Branch

The fixed-length sequence representation ($r^{(s)}$) obtained from scalar attention pooling was passed to a neural prediction branch for multilabel GO term prediction. This branch consisted of a lightweight multilayer perceptron (MLP) that first applied layer normalization to the pooled sequence representation, followed by dropout, a fully connected hidden layer, a GELU activation function, a second dropout layer, and a final linear output layer. A detailed description of the neural prediction branch architecture is seen in Figure 2.

Given the pooled sequence representation $r^{(s)} \in \mathbb{R}^{1280}$, the neural prediction branch produced a vector of logits:

$$z^{(n)} = \text{MLP}(r^{(s)}) \quad (4)$$

Where $z^{(n)} \in \mathbb{R}^C$, and C denotes the number of GO terms for the selected ontology aspect. The logits were transformed into independent GO-term probabilities using a sigmoid activation function:

$$p^{(n)} = \sigma(z^{(n)}) \quad (5)$$

Where $p^{(n)} \in [0,1]^C$ represents the neural branch's predicted probability for each GO term.

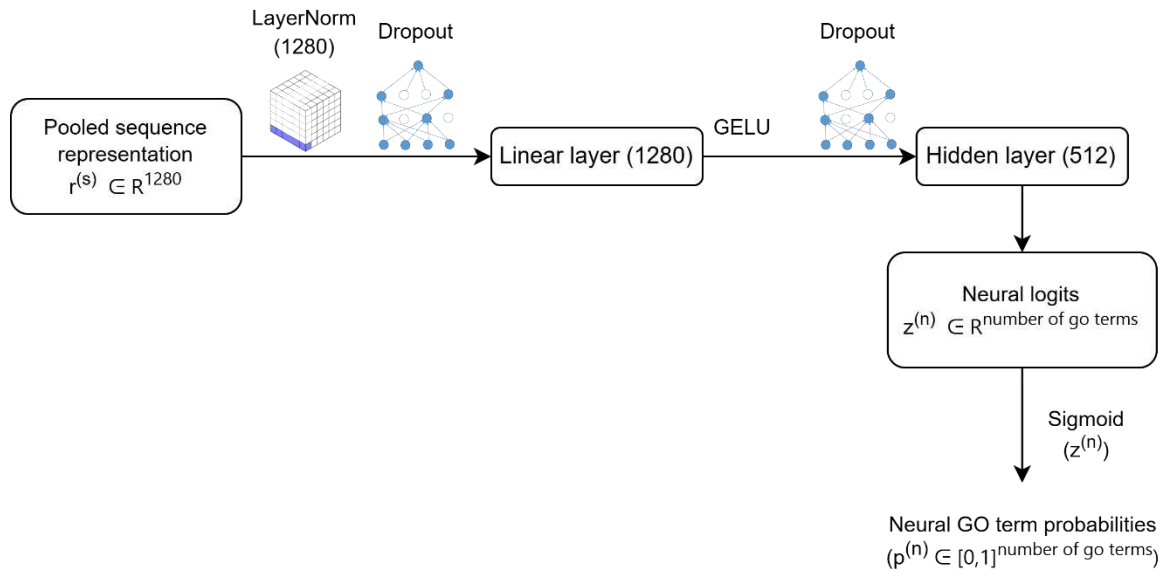


Figure 2. Architectural Diagram of Neural Prediction Branch. *The neural prediction branch was implemented as a two-layer feed-forward MLP with one hidden layer. It consisted of layer normalization, dropout, a hidden fully connected layer, GELU activation, a second dropout operation, and a final fully connected output layer.*

Homology Branch

To incorporate sequence homology information, a DIAMOND-based homology branch was constructed using the PDB training set proteins as the reference database. For the training split, self-hits were removed to prevent direct label copying from the query protein itself. Validation and test proteins were also searched only against the training database using DIAMOND blastp to avoid leakage from validation or test annotations.

DIAMOND search was performed using an E-value threshold of 1×10^{-5} , a minimum query coverage of 0.30, a maximum of 50 target sequences, and sensitive iterative search. After parsing the DIAMOND output, duplicate query-subject alignments were resolved by retaining the best alignment according to bitscore, query coverage, E-value, and percent identity. Hits were then filtered by E-value and query coverage, sorted by alignment strength, and the top $K = 10$ retained hits were used to construct the homology-derived prediction vector.

For each query protein, the retained homologs were converted into a GO-term prior vector. Let $\mathcal{H}$ denote the set of retained homologous hits for a query protein. Each homolog $j \in \mathcal{H}$ was assigned an unnormalized score based on its DIAMOND bitscore and query coverage:

$$\tilde{w}_j = b_j c_j \quad (6)$$

where b_j is the DIAMOND bitscore and c_j is the query coverage for homolog j . The product of bitscore and query coverage was used to emphasize homologs that exhibit both strong alignment quality and broad sequence coverage, while down-weighting matches that are high in only one of these measures. The final normalized homology weight was computed as:

$$w_j = \frac{b_j c_j}{\sum_{m \in \mathcal{H}} b_m c_m + \epsilon} \quad (7)$$

where ϵ is a small constant for numerical stability. For each GO term k , the homology prior was computed as the sum of the normalized weights of retained homologs annotated with that term:

$$p_k^{(h)} = \sum_{j \in \mathcal{H}} w_j \mathbb{I}(k \in A_j), \quad k = 1, \dots, C \quad (8)$$

where A_j denotes the set of GO annotations associated with homolog j , $\mathbb{I}(\cdot)$ is an indicator function, and C is the number of GO terms for the selected ontology aspect. This produced a homology-derived prior vector:

$$p^{(h)} \in [0,1]^C \quad (9)$$

After construction, the prior vector was clipped to the valid probability range:

$$p^{(h)} \leftarrow \text{clip}(p^{(h)}, 0, 1) \quad (10)$$

If no valid homologs were retained for a query protein, the homology prior was set to a zero vector, $p^{(h)} = \mathbf{0} \in \mathbb{R}^C$.

Implementation of Gating Mechanism

To combine sequence-derived neural predictions with homology-derived prediction scores, we used a reliability-aware gating mechanism. The purpose of the gate was to learn, for each protein, how much weight should be assigned to the neural prediction branch and how much weight should be assigned to the homology branch.

Neural and homology branches produced a vector of GO-term probabilities, $p^{(n)}$; $p^{(h)} \in [0,1]^C$. Where C is the number of GO terms for the selected ontology aspect. For each protein, the gate received a four-dimensional reliability feature vector derived from the retained DIAMOND hits:

$$q = [b_{\max}, \text{cov}_{\max}, \log(1 + n_{\text{hits}}), \text{has_hit}]$$

Here, $b_{\max}$ is the maximum alignment bitscore among retained hits, $\text{cov}_{\max}$ is the maximum query coverage, n_{hits} is the number of retained hits, and has_hit is a binary indicator denoting whether any valid homology hit was available.

The gate was implemented as a small neural network consisting of layer normalization, a fully connected hidden layer, a GELU activation, dropout, a final linear layer, and a softmax output. Given the reliability feature vector q , the gate produced two fusion weights:

$$[\alpha_n, \alpha_h] = \text{softmax}(g_{\phi}(q)) \quad (11)$$

where α_n is the weight assigned to the neural prediction branch and α_h is the weight assigned to the homology branch, and $\alpha_n + \alpha_h = 1$.

The final prediction was computed as a convex combination of the neural and homology predictions:

$$p = \alpha_n p^{(n)} + \alpha_h p^{(h)} \quad (12)$$

with the final fused multilabel GO-term probability vector defined as:

$$p \in [0,1]^C$$

The final gate layer was initialized to produce approximately balanced weights, so that the model began training with near-equal reliance on the neural and homology branches. During training, the gate learned to adjust these weights based on the reliability of the available homology evidence. Thus, proteins with stronger or more reliable homology evidence could receive greater contribution from the homology branch, while proteins with weak or missing homology evidence could rely more heavily on the neural sequence-based prediction branch.

Training Objective

The model was trained as a multilabel classification model, where each protein could be associated with multiple Gene Ontology terms within the selected ontology aspect. For each protein, the model produced a probability vector, $\hat{y} \in [0,1]^C$, where C denotes the number of GO terms in the aspect-specific vocabulary.

The ground-truth annotation for each protein was represented as a multi-hot target vector, $y \in \{0,1\}^C$, where $y_j = 1$ indicates that the protein is annotated with GO term j , and $y_j = 0$ otherwise.

To account for the strong class imbalance typical of GO-term prediction, the primary prediction loss was a weighted binary cross-entropy loss computed on the predicted probabilities. For a batch of B proteins and C GO terms, this loss was defined as:

$$\mathcal{L}_{\text{WBCE}} = \frac{1}{BC} \sum_{i=1}^B \sum_{j=1}^C [-w_j y_{ij} \log(\hat{y}_{ij}) - (1 - y_{ij}) \log(1 - \hat{y}_{ij})] \quad (13)$$

where $\hat{y}_{ij}$ is the predicted probability for GO term j in protein i , y_{ij} is the corresponding binary target, and w_j is the positive-class weight for GO term j . The positive-class weights were computed from the training labels as:

$$w_j = \frac{N-n_j}{n_j+1} \quad (14)$$

where N is the number of valid training proteins for the selected ontology aspect, and n_j is the number of training proteins annotated with GO term j . These weights were capped at a maximum value to prevent extremely rare GO terms from producing excessively large loss contributions.

GO Hierarchy Enforcement

To further discourage hierarchy-inconsistent predictions, a soft hierarchy-violation penalty was added to the training objective. Direct child-parent GO term pairs were extracted from the ontology graph for the selected GO aspect and mapped to their corresponding output indices in the model vocabulary. For each predicted probability vector, the model was penalized when the predicted probability of a child term exceeded the predicted probability of its parent term.

Given predicted probabilities:

$$\hat{Y} \in [0,1]^{B \times C}$$

where B is the batch size and C is the number of GO terms for the selected ontology aspect, the hierarchy loss was computed over the set of direct child-parent pairs $\mathcal{E}$. For a child-parent pair $(c, a) \in \mathcal{E}$, c denotes the child GO-term index and a denotes the parent GO-term index.

The hierarchy loss was defined as:

$$\mathcal{L}_{\text{hier}} = \frac{1}{B|\mathcal{E}|} \sum_{b=1}^B \sum_{(c,a) \in \mathcal{E}} \max(0, \hat{y}_{b,c} - \hat{y}_{b,a}) \quad (15)$$

where $\hat{y}_{b,c}$ is the predicted probability of the child GO term for protein b , and $\hat{y}_{b,a}$ is the predicted probability of its parent GO term. This penalty is zero when parent terms have probabilities greater than or equal to their child terms, and increases when child terms are

assigned higher probabilities than their parents. If no child-parent pairs were available for the selected GO vocabulary, the hierarchy loss was set to zero.

The final training objective combined the weighted binary cross-entropy (BCE) loss with the hierarchy penalty:

$$\mathcal{L} = \mathcal{L}_{\text{BCE}} + \lambda_{\text{hier}} \mathcal{L}_{\text{hier}} \quad (16)$$

where λ_{hier} controls the strength of the hierarchy regularization. This formulation treated hierarchy enforcement as a soft constraint rather than a hard post-processing rule. Therefore, predictions were not forcibly modified after inference; instead, the model was trained to reduce violations in which a child GO term received a higher probability than its parent.

During training, the loss was computed for each batch, gradients were backpropagated through the trainable components of the model, and parameters were updated using the optimizer. Validation used the same loss decomposition, reporting the total validation loss together with the weighted BCE component and hierarchy loss component.

Evaluation metrics

We used the standard CAFA metrics [27] to evaluate EIHR-PROT and other baseline methods on the PDB test set.

Fmax is the maximum F1 score obtained across all prediction-score thresholds. It measures the balance between precision and recall.

Let T_i denote the propagated ground-truth GO-term set for protein i , and let $P_i(\tau)$ denote the propagated predicted set obtained by thresholding the model score at τ .

Per-protein precision is defined as:

$$\text{Precision}_i(\tau) = \frac{\text{count}(P_i(\tau) \cap T_i)}{\text{count}(P_i(\tau))} \quad (17)$$

337 Per-protein recall is defined as:

$$338 \quad \text{Recall}_i(\tau) = \frac{\text{count}(P_i(\tau) \cap T_i)}{\text{count}(T_i)} \quad (18)$$

339 The implementation averages precision over proteins with at least one predicted term at the
340 threshold, and recall over all evaluated proteins. The F-score at threshold τ is then:

$$341 \quad F(\tau) = \frac{2 \cdot P(\tau) \cdot R(\tau)}{P(\tau) + R(\tau)} \quad (19)$$

342 The reported Fmax is the maximum F-score across thresholds:

$$343 \quad Fmax = \max_{\tau} F(\tau) \quad (20)$$

344 AUPR is a common metric for evaluating for highly-imbalanced datasets. It is computed as the
345 area under the precision–recall curve after sorting the evaluation points by recall.

$$346 \quad AUPR = \int \text{Precision}(\text{Recall}) d(\text{Recall}) \quad (21)$$

347 Smin represents the semantic distance between predicted and real annotations considering
348 information content of each function. Information content (IC) is estimated from the
349 propagated training annotations. For each GO term g , IC is computed from the ratio of the
350 minimum direct-parent count to the term count:

$$351 \quad IC(g) = \log_2\left(\frac{\min \text{parent count}(g)}{\text{count}(g)}\right) \quad (22)$$

352 Using this IC weighting, Remaining Uncertainty (RU) and Misinformation (MI) are defined
353 for each protein at threshold τ as:

$$354 \quad RU_i(\tau) = \sum_{g \in T_i \setminus P_i(\tau)} IC(g) \quad (23)$$

$$355 \quad MI_i(\tau) = \sum_{g \in P_i(\tau) \setminus T_i} IC(g) \quad (24)$$

356 These quantities are averaged across evaluated proteins:

$$RU(\tau) = \frac{1}{N} \sum RU_i(\tau) \quad (25)$$

$$MI(\tau) = \frac{1}{N} \sum MI_i(\tau) \quad (26)$$

The semantic distance at threshold τ is:

$$S(\tau) = \sqrt{RU(\tau)^2 + MI(\tau)^2} \quad (27)$$

S_{min} is the minimum semantic distance across thresholds.

$$S_{min} = \min_{\tau} S(\tau) \quad (28)$$

Model Training and Evaluation

All trainable models were implemented in PyTorch (version 2.11) and trained using the AdamW optimizer. For each Gene Ontology aspect, a separate output vocabulary was constructed from the training annotations, and model training was performed independently. Hyperparameters were selected using randomized search over predefined search spaces. For each sampled hyperparameter configuration, the model was trained for 20 epochs and evaluated on the validation set. The best-performing configurations, ranked by validation Fmax, were then used for final model training. Details on the hyperparameter search space for the randomized search and the values used for final model training are provided in Supplementary Table S5.

During final training, each candidate configuration was trained for up to 120 epochs with early stopping. All experiments were run using a batch size of 16. Training was performed on an NVIDIA RTX4090 24GB and metrics were tracked using Weights & Biases.

Baseline Methods

To contextualize the performance of the proposed model, we compared it against four baseline methods: a naïve model, DIAMOND Score, DeepGOPlus, and TALE+. DeepGOPlus and TALE+ provide an external reference for combined sequence and homology-based inference methods. All baseline models were evaluated with the PDB and AF test sets. DeepGOPlus and TALE+ were retrained on the PDB training set before conducting evaluations, while dataset-specific preprocessing, GO vocabulary construction, and CAFA-style evaluation were adapted to ensure a controlled comparison.

While more recent methods such as TEMPROT+ have been published, their computational requirements preclude straightforward implementation and comparison within our available computational resources. Furthermore, the focus of this work is not to establish a new state-of-the-art benchmark but to demonstrate that combining predictions from multiple modalities using a gating mechanism based on established reliability signals, a principle that could be integrated into any hybrid architecture, improves performance over existing fusion approaches. We chose DeepGOPlus and TALE+ as baselines because they represent well-established, widely used methods that, like EIHR-PROT, combine sequence-based neural predictions with homology-derived evidence.

Naïve Approach

We used a simple baseline to assign the same GO terms to every protein according to their frequency in the training set. Because GO is hierarchical and annotations are propagated, high-level terms often end up assigned to many proteins. This naïve approach is commonly used as a baseline for evaluating protein function annotation models.

For a query protein p , the score for a GO term f is given by:

$$S(p, f) = \frac{N_f}{N_{total}} \quad (29)$$

Here, N_f is the number of training proteins annotated with f , and N_{total} is the total number of proteins in the training set.

DIAMOND Score

DIAMOND Score identifies similar sequences from a training database and transfers annotations from highly similar hits. Unlike EIHR-PROT (w/o sequence branch), our homology-only ablation, DIAMOND Score normalizes the total bitscore from similar sequences. The prediction score for a GO term is computed from the weighted contribution of DIAMOND matches as shown below.

$$S(q, f) = \frac{\sum_{s \in E} \text{bitscore}(q, s) \cdot I(f \in T_s)}{\sum_{s \in E} \text{bitscore}(q, s)} \quad (30)$$

where, E denotes the set of similar sequences, $\text{bitscore}(q, s)$ is the DIAMOND bitscore between query q and sequence s , T_s is the annotation set of s , and $I(\cdot)$ equals 1 when f is present in T_s and 0 otherwise.

DeepGOPlus

DeepGOPlus is a sequence-based protein function prediction method that combines a one-dimensional convolutional neural network with sequence similarity-based inference. Its neural component represents amino acid sequences using one-hot encoding and applies multiple 1D convolutional filters of different lengths to detect sequence motifs associated with protein function. The neural prediction is then combined with DIAMOND-based homology scores using a weighted sum.

TALE+

TALE is a Transformer-based protein function annotation method that jointly embeds protein sequences and GO labels in a shared latent space. It uses self-attention to capture global dependencies in protein sequences and explicitly models label information using the

hierarchical structure of Gene Ontology. TALE+ extends TALE by combining predictions from an ensemble of TALE models with DIAMOND-based sequence similarity scores.

Results

EIHR-PROT Improves Protein Function Annotation Compared to Baseline Models

To evaluate the performance of EIHR-PROT, we compared the model against other baseline methods, including Blast and a Naïve predictor, which predicts only the k highest frequency GO terms across all aspects. We also evaluated EIHR-PROT against other state of the art methods that infuse protein sequences and homology priors for functional annotation such as DeepGOPlus [21] and TALE+ [22]. These models were compared using the complete PDB test set and the results are provided in Table 1.

For all BP, CC, and MF prediction tasks, EIHR-PROT achieved Fmax scores of 0.6039, 0.6755, and 0.7319 respectively; AUPR scores of 0.6015, 0.6940, and 0.7002; and Smin scores of 66.8015, 16.3879, and 9.9753. The model outperformed all baseline methods across all metrics, except Smin, where DeepGOPlus showed better output for BP and CC annotation tasks.

In contrast to DeepGOPlus and TALE+, EIHR-PROT offers a more flexible modality weighting and reduces the impact of noisy GO term predictions from the homology branch, leading to more accurate predictions. These performance gains were most pronounced for CC GO terms where Fmax and AUPR improved by 0.05 and 0.1 compared to the nearest model.

Ablation Study

To adequately ascertain the impact of the gating mechanism and other model framework components, we performed ablation experiments. For these experiments, we evaluated three models:

- (i) EIHR-PROT (w/o explicit homology reliability signals): Explicit homology signals were removed and the model was re-trained to enable the gating mechanism learn appropriate weights for sequence and homology branches internally.
- (ii) EIHR-PROT (w/o homology priors): ESM-2 embeddings were extracted, pooled, and passed to the neural prediction branch without fusion with homology priors.
- (iii) EIHR-PROT (w/o sequence branch): Protein sequences were evaluated against the DIAMOND database to produce homology prior probability scores without fusion with ESM-based predictions.

The results of this evaluation are provided in Table 2. Our ablation studies demonstrated that removing the explicitly incorporated homology reliability signals decreased model performance across all metrics for BP, CC, and MF. This impact was strongest for BP terms, indicating that unreliable homology terms have the greatest impact for BP term prediction in fused sequence-homology methods, compared to other aspects. Because EIHR-PROT uses different learned weights per protein, rather than learning a fixed α_n and α_h for all proteins per ontology aspect, the model can down-weight noisy homology priors at a protein-specific level. Removing the sequence branch produced the most drastic drop in performance across all ontologies and metrics. EIHR-PROT (w/o explicit homology reliability signals) also performed poorly for Fmax, AUPR, and Smin, compared to EIHR-PROT (w/o homology priors), except for CC where Fmax improved marginally with the implicit gating mechanism. We hypothesize that uncontrolled fusion of sequence with direct homology signals may dilute the evolutionary context already captured by PLM representations and reduce overall predictive performance. Therefore, our model provides a method for maximizing the benefits of added homology prior context without diluting the strength of the signals from the sequence representation branch. Comparisons of ablations across each of the various homology bins are available in Supplementary Table S2.

***Table 1: Comparison of Baseline Methods of Models on the PDB Test Set**

Model	<u>F_{max}</u>			AUPR			<u>S_{min}</u>		
	BP	CC	MF	BP	CC	MF	BP	CC	MF
Naïve	0.3751	0.4982	0.3218	0.3063	0.2395	0.3941	96.7288	25.1695	20.3617
DIAMOND Score	0.5189	0.4211	0.6729	0.0735	0.0863	0.0763	84.4520	23.2809	12.3445
EIHR-PROT	0.6039	0.6755	0.7310	0.6015	0.6940	0.7002	66.8015	16.3879	9.9753
<u>DeepGOPlus</u>	0.5890	0.6290	0.7090	0.5240	0.5990	0.6880	60.6430	14.1560	10.101
TALE+	0.5849	0.6208	0.6755	0.4239	0.4554	0.4046	239.5941	34.7171	23.4236

*Figures in bold in each column represent the best performance for that metric

***Table 2: Ablation Studies of EIHR-PROT on the PDB Test Set**

Model	$\underline{F_{max}}$			AUPR			$\underline{S_{min}}$		
	BP	CC	MF	BP	CC	MF	BP	CC	MF
EIHR-PROT	0.6039	0.6755	0.7310	0.6015	0.6940	0.7002	66.8015	16.3879	9.9753
EIHR-PROT (w/o explicit homology reliability signals)	0.5107	0.6587	0.673	0.5273	0.6621	0.6596	72.9757	16.9390	11.1769
EIHR-PROT (w/o homology priors)	0.5809	0.6556	0.7172	0.5704	0.6633	0.7542	71.7139	16.9089	10.0555
EIHR-PROT (w/o sequence branch)	0.5188	0.4211	0.6730	0.0698	0.0839	0.0701	72.5894	19.7890	11.1896

*Figures in bold in each column represent the best performance for that metric

Adaptation of Gating Mechanism Based on Sequence Similarity

We measured the changes in mean homology weights produced by the EIHR-PROT and EIHR-PROT (w/o explicit homology reliability signals). These comparisons were performed to determine if the gate accurately down-weights homology priors in low-homology settings. The PDB test set was split into five thresholds (<30%, <40%, 50%, <70%, and <95%). These thresholds represent the degree of similarity to the training set, which was used to construct the homology query database. Figure 3 shows that EIHR-PROT consistently reduces homology prior weights as homology trustworthiness reduces. Gate weights for the homology branch in EIHR-PROT reduced by 0.094, 0.068, and 0.091 for BP, CC, and MF aspects between <95% and 30% similarity bins. In contrast, EIHR-PROT (w/o explicit homology reliability signals) demonstrated negligible change in gate weights based on the strength of homology signals. Examination of changes in BP gate weights revealed a reduction of 0.011 while CC and MF aspects increase homology weighting by 1.5×10^{-3} and 3.5×10^{-5} , respectively. Mean and

standard deviation of gate weights for EIHR-PROT and EIHR-PROT (w/o explicit homology reliability signals) are available in Supplementary Table S3.

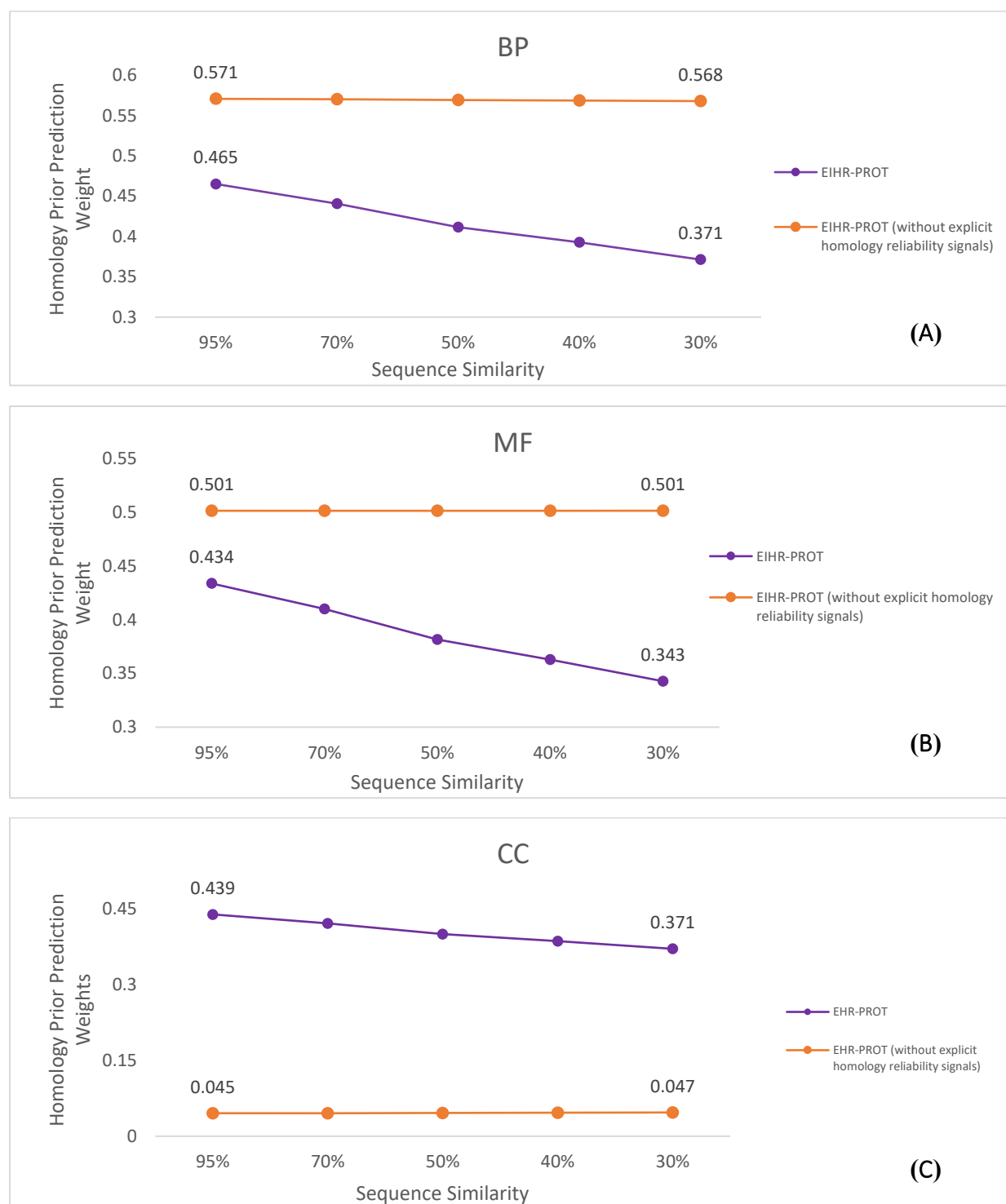


Figure 3: Changes in homology gate weights across different sequence similarity bins on the PDB test set. A, B, and C represent the change in homology prediction weights for BP,

CC, and MF terms respectively. Homology weights showed a steady decline as sequence similarity trustworthiness reduces.

Model Performance on External Dataset Baseline

We further examined the robustness of EIHR-PROT by testing its performance on the AlphaFold test set (AF test set). This dataset, curated by Gu et al. [28], represents a low-homology protein dataset comprising 567 protein sequences that share less than 25% sequence identity with the PDB training set obtained using the MMseqs2 tool. This evaluation was conducted to examine the performance of EIHR-PROT compared to other combined sequence and homology-based frameworks for novel sequences with highly limited annotation priors. As shown in Figure 4, EIHR-PROT outperforms DeepGOPlus and TALE+ across all metrics, except BP and CC, where DeepGOPlus shows improved performance in Smin.

DeepGOPlus's better Smin performance is likely attributed to the distinct objectives captured by these evaluation metrics. Whereas Fmax and AUPR primarily assess classification accuracy, Smin additionally weights prediction errors according to the information content of GO terms, making it particularly sensitive to semantically costly false-positive and false-negative predictions. EIHR-PROT was optimized to maximize multilabel prediction accuracy through learned-gating-based fusion rather than directly minimizing semantic distance.

Consequently, our adaptive fusion strategy may improve overall annotation performance while occasionally introducing prediction sets with marginally higher semantic penalties. Furthermore, DeepGOPlus's fixed integration of sequence similarity information may yield more conservative predictions in some cases, reducing semantically expensive errors despite lower overall predictive accuracy. These findings suggest that maximizing classification performance and minimizing semantic distance are related but not identical objectives. Fmax,

AUPR, and Smin scores for EIHR-PROT and baseline methods are provided in Supplementary Table S4.

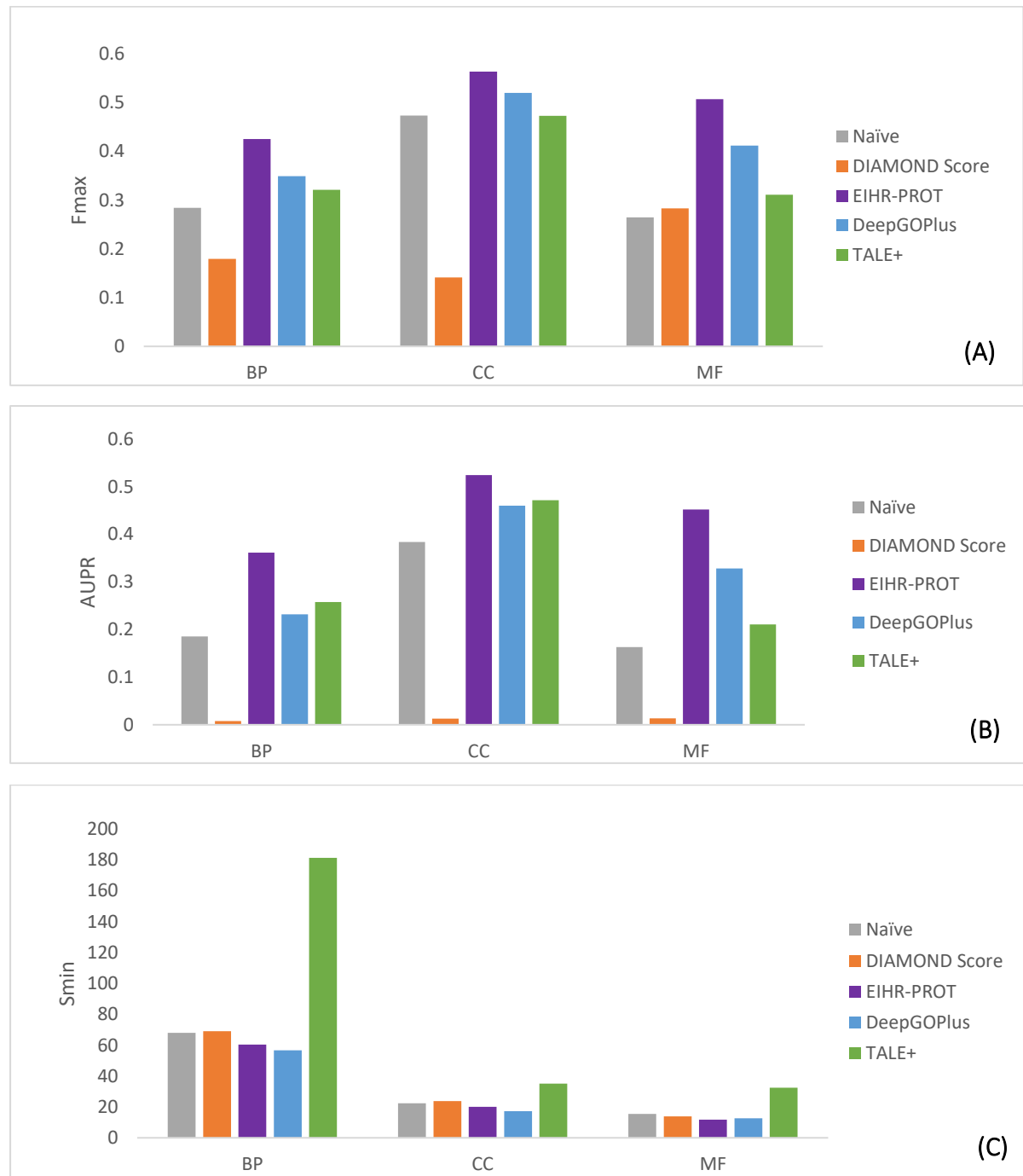


Figure 4. Comparison of Baseline methods on the Aftest set. *A, B, and C represent the performance of EIHR-PROT compared with baseline methods across Fmax, AUPR, and Smin CAFA metrics respectively.*

Discussion

530 EIHR-PROT combines learned conserved motifs, implicit secondary structure information,
531 and other evolutionary patterns captured by protein language model embedding representations
532 with direct functional annotation transfer through homology.

533 Compared to DeepGOPlus and TALE+, our model computes homology branch weights at a
534 per-protein level. This approach makes EIHR-PROT robust, even in low-homology settings.

535 The current iteration of our model is simple to use and does not require experimentally derived
536 protein structures, network information, or other data for performing inference. In the absence
537 of hits to our training database, EIHR-PROT's sequence branch remains a highly competitive
538 protein function annotation tool. However, while the experiments showed that the sequence
539 branch provided the strongest contribution to the model's performance across low homology
540 bins, low-weighted homology predictions still offered valuable information and improved
541 performance, justifying the choice of our hybrid model. Furthermore, EIHR-PROT provides
542 greater interpretability, allowing researchers to understand how much each branch contributed
543 to the final prediction value for each protein, in addition to the overall probability score for
544 each GO term prediction.

545 Our results indicate that the effectiveness of hybrid protein function prediction depends not
546 only on the quality of the individual predictors but also on accurately estimating when each
547 predictor should be trusted. This suggests that future improvements may arise as much from
548 principled evidence integration as from increasingly complex prediction architectures.

549 One limitation we must highlight is that EIHR-PROT truncates sequences longer than 1022
550 amino acids. This is due to the current limitations of the ESM-2 protein language model, where
551 we obtain the sequence-branch's amino acid representations. We expect that the increase of
552 high-quality protein annotation data and future improvements in protein language model
553 representations, will enable EIHR-PROT to improve its performance.

Later research efforts will explore incorporating other modalities, such as structure, protein-protein interaction (PPI), and genomic context, along with their reliability metrics into our gated architecture. The goal of these future studies will be to determine if the gating mechanism will maintain its ability to effectively learn modality weights as the modalities increase. Also, we will evaluate whether incrementing the number of modalities will yield improved performance and if these potential gains will be meaningful enough to justify the added architectural complexity and compute overhead.

Conclusion

This study demonstrates that explicitly modelling the reliability of homology evidence can improve the effectiveness of hybrid protein function prediction. By allowing the contribution of sequence-based and homology-based predictions to vary according to the strength of the available sequence-similarity evidence, EIHR-PROT achieved consistently higher predictive performance than comparable sequence-homology methods across all Gene Ontology aspects evaluated. The accompanying ablation studies and gate-weight analyses further showed that these improvements were attributable to the direct incorporation of homology reliability signals, enabling the model to appropriately reduce reliance on homology information when sequence similarity was weak while still benefiting from annotation transfer when reliable homologues were available.

These findings highlight that the success of hybrid protein function prediction models depends not only on combining complementary sources of biological information, but also on estimating how much confidence should be placed in each source for individual proteins. Rather than treating homology as uniformly informative, EIHR-PROT demonstrates the value of adapting evidence integration to the reliability of the available biological context. This

577 provides a more robust and interpretable approach to protein function annotation, particularly
578 for proteins spanning heterogeneous sequence-similarity regimes.

579 Overall, EIHR-PROT contributes a simple yet effective framework for adaptive evidence
580 integration in protein function prediction. Beyond the specific architecture presented here, the
581 results suggest that learned gating based fusion represents a promising direction for future
582 computational annotation methods and may provide a flexible foundation for incorporating
583 additional biological modalities as high-quality functional data continue to expand.

584 **Declarations**

585 **Ethics approval and consent to participate:** Not Applicable

586 **Consent for publication:** Not Applicable

587 **Availability of data and materials:**

588 Publicly available in a repository:

589 Protein Databank (PDB) and AlphaFold (AF) secondary datasets used in the study are publicly
590 available and can be accessed at the following URL:
591 <https://github.com/ZhonghuiGu/HEAL/tree/main/data>

592 Final model checkpoints for EIHR-PROT are stored in Zenodo,
593 <https://doi.org/10.5281/zenodo.20661853>.

594 **Competing interests:** The authors declare that they have no competing interests

595 **Funding:** The authors have no external funding to declare.

Authors' contributions: OJ designed the study, proofread the manuscript, and supervised the model training and evaluation. UE wrote the code and manuscript and performed model training and evaluation.

Acknowledgements: Authors would like to acknowledge the faculty and staff of the Department of Computer and Information Sciences, Covenant University for their contributions to this manuscript.

References

1. Rives A, Meier J, Sercu T, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl Acad Sci U S A*. 2021;118(15):e2016239118. doi:10.1073/pnas.2016239118.
2. Elnaggar A, Heinzinger M, Dallago C, et al. ProtTrans: Toward understanding the language of life through self-supervised learning. *IEEE Trans Pattern Anal Mach Intell*. 2022;44(10):7112-27. doi:10.1109/TPAMI.2021.3095381.
3. Littmann M, Heinzinger M, Dallago C, et al. Embeddings from deep learning transfer GO annotations beyond homology. *Sci Rep*. 2021;11:1160. doi:10.1038/s41598-020-80786-0.
4. Heinzinger M, Elnaggar A, Wang Y, Dallago C, Nechaev D, Matthes F, et al. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinformatics*. 2019;20(1):723. doi:10.1186/s12859-019-3220-8.
5. Villegas-Morcillo A, Makrodimitris S, van Ham RCHJ, Gomez AM, Sanchez V, Reinders MJT. Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function. *Bioinformatics*. 2021;37(2):162-70. doi:10.1093/bioinformatics/btaa768.

- 619 6. Lai B, Xu J. Accurate protein function prediction via graph attention networks with
620 predicted structure information. *Brief Bioinform.* 2022;23(1):bbab502.
621 doi:10.1093/bib/bbab502.
- 622 7. Gligorijević V, Renfrew PD, Kosciulek T, Leman JK, Berenberg D, Vatanen T, et al.
623 Structure-based protein function prediction using graph convolutional networks. *Nat*
624 *Commun.* 2021;12(1):3168. doi:10.1038/s41467-021-23303-9.
- 625 8. You R, Yao S, Mamitsuka H, Zhu S. DeepGraphGO: Graph neural network for large-
626 scale, multispecies protein function prediction. *Bioinformatics.* 2021;37(Suppl 1):i262-
627 i271. doi:10.1093/bioinformatics/btab270.
- 628 9. You R, Huang X, Zhu S. DeepText2GO: Improving large-scale protein function
629 prediction with deep semantic text representation. *Methods.* 2018;145:82-90.
630 doi:10.1016/j.ymeth.2018.05.026.
- 631 10. Kulmanov M, Guzmán-Vega FJ, Roggli PD, et al. Protein function prediction as
632 approximate semantic entailment. *Nat Mach Intell.* 2024;6(2):220-8.
633 doi:10.1038/s42256-024-00795-w.
- 634 11. Piovesan D, Giollo M, Ferrari C, Tosatto SC. Protein function prediction using guilty
635 by association from interaction networks. *Amino Acids.* 2015;47(12):2583-92.
636 doi:10.1007/s00726-015-2049-3.
- 637 12. Ma W, Bi X, Jiang H, et al. Annotating protein functions via fusing multiple biological
638 modalities. *Commun Biol.* 2024;7:1705. doi:10.1038/s42003-024-07411-y.
- 639 13. Zhou N, Jiang Y, Bergquist TR, Lee AJ, Kacsoh BZ, Crocker AW, et al. The CAFA
640 challenge reports improved protein function prediction and new functional annotations
641 for hundreds of genes through experimental screens. *Genome Biol.* 2019;20(1):244.
642 doi:10.1186/s13059-019-1835-8.

14. Hartout P, Chen D, Pellizzoni P, Oliver C, Borgwardt K. Endowing protein language models with structural knowledge. *Bioinformatics*. 2025;41(11):btaf582. doi:10.1093/bioinformatics/btaf582.
15. Yuan Q, Xie J, Xie J, Zhao H, Yang Y. Fast and accurate protein function prediction from sequence through pretrained language model and homology-based label diffusion. *Brief Bioinform*. 2023;24(3):bbad117. doi:10.1093/bib/bbad117.
16. Chen JY, Wang JF, Hu Y, Li XH, Qian YR, Song CL. Evaluating the advancements in protein language models for encoding strategies in protein function prediction: a comprehensive review. *Front Bioeng Biotechnol*. 2025;13:1506508. doi:10.3389/fbioe.2025.1506508.
17. Zhu YH, Zhang C, Yu DJ, Zhang Y. Integrating unsupervised language model with triplet neural networks for protein gene ontology prediction. *PLoS Comput Biol*. 2022;18(12):e1010793. doi:10.1371/journal.pcbi.1010793.
18. Dai X, Xu Z, Liang Z, Tu X, Zhong S, Schnable JC, et al. Non-homology-based prediction of gene functions in maize (*Zea mays* ssp. *mays*). *Plant Genome*. 2020;13(2):e20015. doi:10.1002/tpg2.20015.
19. Jiang Y, Oron TR, Clark WT, Bankapur AR, D'Andrea D, Lepore R, et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biol*. 2016;17(1):184. doi:10.1186/s13059-016-1037-6.
20. Sanderson T, Bileschi ML, Belanger D, Colwell LJ. ProteInfer, deep neural networks for protein functional inference. *eLife*. 2023;12:e80942. doi:10.7554/eLife.80942.
21. Kulmanov M, Hoehndorf R. DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics*. 2020;36(2):422-9. doi:10.1093/bioinformatics/btz595.

22. Cao Y, Shen Y. TALE: Transformer-based protein function annotation with joint sequence-label embedding. *Bioinformatics*. 2021;37(18):2825-33. doi:10.1093/bioinformatics/btab198.
23. Zhang C, Liu Q, Freddolino L. StarFunc: Fusing template-based and deep learning approaches for accurate protein function prediction. *Genomics Proteomics Bioinformatics*. 2026;qzag018. doi:10.1093/gpbjnl/qzag018.
24. Zhapa-Camacho F, Tang Z, Kulmanov M, Hoehndorf R. Predicting protein functions using positive-unlabeled ranking with ontology-based priors. *Bioinformatics*. 2024;40(Suppl 1):i401-i409. doi:10.1093/bioinformatics/btae237.
25. DeepFRI [github](https://github.com/flatironinstitute/DeepFRI/tree/master/preprocessing/data) repository. <https://github.com/flatironinstitute/DeepFRI/tree/master/preprocessing/data>. Accessed 19 April 2026.
26. Lin Z, Akin H, Rao R, Hie BL, Zhu Z, Lu W, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*. 2023;379(6637):1123-30. doi:10.1126/science.ade2574.
27. Radivojac P, Clark WT, Oron TR, et al. A large-scale evaluation of computational protein function prediction. *Nat Methods*. 2013;10(3):221-7. doi:10.1038/nmeth.2340.
28. Gu Z, Luo X, Chen J, Deng M, Lai L. Hierarchical graph transformer with contrastive learning for protein function prediction. *Bioinformatics*. 2023;39(7):btad410. doi:10.1093/bioinformatics/btad410.

Supplementary Files

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

- [SupplementaryMaterials.docx](#)