FAPM: Functional Annotation of Proteins using Multi-Modal Models Beyond Structural Modeling

Wenkai Xiang[1, 2, #], Zhaoping Xiong[3, #, *], Huan Chen[4, #], Jiacheng Xiong[1,5], Wei Zhang[1,5], Zunyun Fu[1], Mingyue Zheng[1,2,5], Bing Liu[4, *] and Qian Shi[2, *]

[1] Drug Discovery and Design Center, State Key Laboratory of Drug Research, Shanghai Institute of Materia Medica, Chinese Academy of Sciences, 555 Zuchongzhi Road, Shanghai 201203, China.

[2] Lingang Laboratory, Shanghai 200031, China

[3] ProtonUnfold Technology Co., Ltd, Suzhou 215000, China.

[4] BioBank, The First Affiliated Hospital of Xi'an Jiaotong University, Yanta District, Xi'an 710061, China

[5] University of Chinese Academy of Sciences, 19A Yuquan Road, Beijing 100049, China.

* Correspondence author.

Email address: shiqian@lglab.ac.cn (Qian Shi) and bliu2018@xjtu.edu.cn (Bing Liu)

# These authors contributed equally to this work.

Abstract

Assigning appropriate property labels, such as functional terms and catalytic activity, to proteins, remains a significant challenge, particularly for the non-homologous ones. In contrast to prior approaches that mostly focused on protein sequence features, we employ pretrained protein language model to encode the sequence features, and natural language model for the semantic information of property descriptions. Specifically, we present FAPM, a contrastive model between natural language and protein sequence language, which combines the pretrained protein sequence model with the pretrained large language model to generate labels such as GO functional terms and catalytic activity prediction in the format of natural language. Our result shows that FAPM has learned superior representations related to protein properties, outperforming protein sequence-based or structure-based models. Our model achieves state-of-the-art performance both on public benchmarks and on experimentally-annotated phage proteins that have limited known homologous sequences. Furthermore, we demonstrate the flexibility of our model to accept additional free text prompts as input, such as easily accessible taxonomy information, not only enhances its predictive performance but also improves the model's explainability. Our methodology presents a novel avenue for exploration, characterized by its potential to supersede current methodologies reliant on multiple sequence alignment for protein annotation.

Key words: protein annotation, multi-modal model, large language model, functional text generation

Introduction

Proteins are essential components of cells and tissues, as they play key roles in various biological processes that sustain life. Understanding their functions is vital for decoding the molecular intricacies of biological systems and curing disease. Acquiring protein sequences from natural sources is generally straightforward, meanwhile, advancements in computational techniques, such as AlphaFold[1], have significantly improved the precision of predicting their three-dimensional structures[2]. However, only fewer than 1% of the proteins, are accompanied by reviewed functional annotations[3] in global sequence databases such as UniProt[4]. The majority that remains, with their functions still need to be experimentally characterized, are subject to high costs and logistical challenges. Furthermore, proteins demonstrate intricate variability and interactions as parts of their inherent complexity for their experimental validations. Therefore, predicting protein functions, while still challenging due to their complexity, is an important approach to elucidate their roles in pioneering studies.

Currently, protein functions can be described using the Gene Ontology (GO) [5], which is recognized as one of the most successful ontologies in biology. GO encompasses three sub-ontologies: molecular functions (MFO) characterizing the functions of individual proteins, biological processes (BPO) in which proteins participate, and cellular components (CCO) where proteins are active. The GO system facilitates a standardized approach to catalog the vast diversity of protein activities across different organisms. And the functions of these proteins are initially obtained from experimental evidences that published in the scientific literatures. These findings, once validated, are curated by database administrators into annotations within databases like UniProtKB/Swiss-Prot[3], ensuring that the information is accessible and standardized according to GO terms. Currently, the UniProtKB/Swiss-Prot database, known for its manually curated entries, encompasses GO annotations for over 560,000 proteins across thousands of organisms.

Various protein sequence-based methods have been proposed for predicating functions of proteins. As early in 1990, tools like BLAST[6] were developed, which allow researchers to compare an unknown protein sequence against a database of sequences with known annotations to find potential sequence similarities. The underlying assumption is that homologous sequences (those sharing a common evolutionary origin) will be of similar functions. Later, signature-based approaches[7-9] were introduced, which catalog domains and motifs within the proteins with similar functions based on scoring against large databases of statistical models for each sequence family with experimentally verified functions. These methods are now employed to update widely-used databases like InterPro[10]. Despite their significant impact, these computational modeling approaches primarily rely on sequence similarity for prediction, leaving a considerable number of proteins unannotated.

Recent research has introduced a variety of deep neural network models to harvest multiple data simultaneously for functional predictions of protein, leveraging diverse sources of information, including sequence data, interactions and domain annotations. For example, DeepGOPlus[11], DeepGOZero[12] and DeepGraphGO[13] utilize domain features extracted from InterPro to train a classification model. In particular, DeepGraphGO uses graph convolutional layers[14] to capture neighboring information within the protein-protein interaction (PPI[15]) network extracted from STRING[16]. Inspired by the success of convolutional neural networks(CNN)[17] in image,

DeepGOCNN[11], DeepGOPlus and ProteInfer[18] utilize convolutional layers in CNN to extract features from protein sequences. Notably, ProteInfer inputs each amino acid after one-hot coding into the CNN network structure. Transformers[19] have achieved great success in many artificial intelligence fields[20, 21]. Recently, it has also been discovered to have potential in protein modeling. ProteinBert[22] utilises protein sequences and semantic features of GO tags for pre-training. GoProFormer[23] combines sequence transformer and graph transformer to learn sequence features and GO network features for predicting GO tags. ESM-2 (Evolutionary Scale Modeling-2)[24] learns patterns and properties in protein sequences through pre-training on a large number of unlabeled protein sequences. These trained models are then applied to specific tasks of predicting protein structure and properties using fine-tuning and transfer learning, leading to improved performance in biomedical research, drug design, and other related fields[25-27]. Many large language models (LLMs) are capable of multitasking, including tasks related to proteins. For example, Galactica[28] is a LLM trained on scientific knowledge corpus, which can be applied to multi-modal tasks involving SMILES chemical formulas and protein sequences. And Mol-Instructions[29] combine construction methods of self-instruct, template-based conversion, and human-crafted task descriptions to cover a wider range of biomolecular tasks. The methods mentioned above primarily depend on sequence similarity or sequence feature modeling. However, the semantic information inherent in the functional description of proteins is rarely used, which is key to establish a comprehensive sequence-to-function link.

Here, we use a contrastive learning framework to jointly model protein sequences and functional texts in our model. To be specific, we consider protein sequences as a modality, and the natural language of functional annotations as another modality. We align the protein sequence modality to natural language modality using contrastive learning approach. This leads to our framework FAPM (**F**unctional **A**nnotation of **P**roteins using **M**ulti-modal models), a contrastive learning method that generates functional descriptions for proteins by leveraging a pre-trained protein sequence model and large language model. FAPM efficiently harnesses the generative capabilities of language models to articulate and generate the functional-semantic information inherent in protein sequences. The notable features of our work are as follows:

(1) Through the integration of multi-modal modeling, we effectively combine protein sequence modeling with language generation frameworks. The analysis in "the explainability of multi-modal representation" section shows the natural connection between the learned representations and their property descriptions, allowing for the direct generation of accurate annotations.
(2) Comparative evaluations on the Swiss-Prot dataset and dozens of phage proteins demonstrate FAPM's superior predictive performance over existing methods, underscoring the model's effectiveness in accurately predicting protein functions.
(3) Optional information, such as taxonomy, can be used as the additional input for the model to improve the generation quality, demonstrating a more flexible way to annotate proteins.

Materials and Methods

FAPM utilizes a Querying Transformer (Q-Former) to address the modality gap in

frozen pre-trained unimodal models. Q-Former are pre-trained in two stages: (1) protein-language representation learning stage with a frozen protein encoder and (2) protein-to-language generative learning stage with a frozen Large Language Model (LLM).

Methods

Model Architecture overview

Figure 1 shows a schematic procedure of FAPM, two frozen pre-train models connected by Q-Former. Specially, we use ESM2 for protein sequence encoding and Mistral-7B[30] for text generation due to their outstanding performance. Q-Former, initialized with pre-trained weights of BERTbase[31], comprises transformer modules for extracting features from the frozen protein model and encoding/decoding the text. The total learnable parameters in Q-Former amount to approximately 188 million.

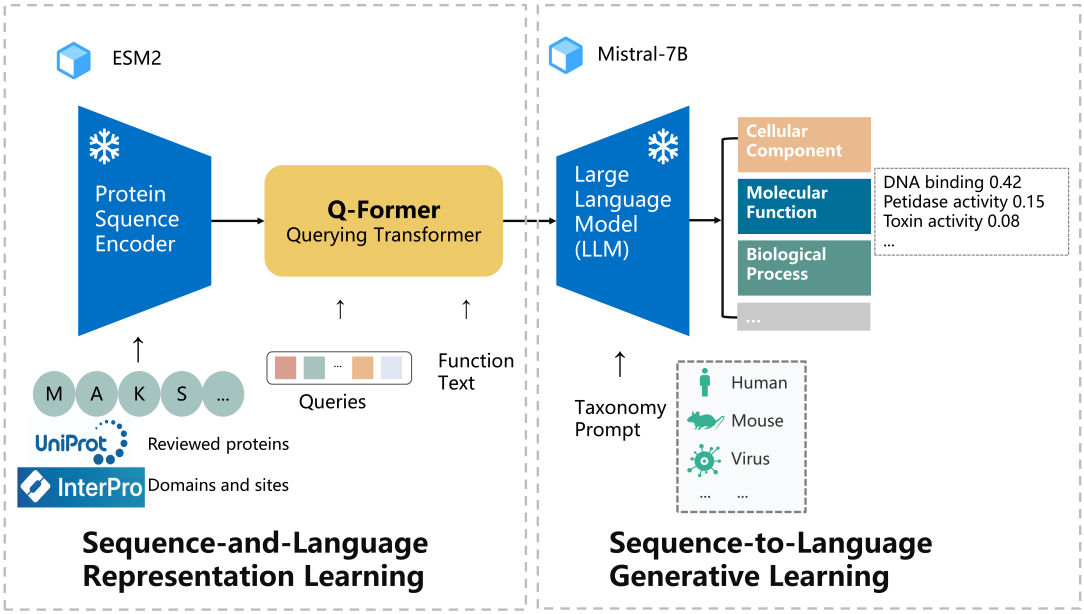

Figure 1 The overview of the framework of FAPM. In the Sequence-and-Language Representation Learning stage, FAPM use ESM2 to encode protein sequence and Q-Former to align them with function text (GO terms). In the Sequence-to-Language Generative Learning stage, Mistral-7B processes the learned queries from the prior stage, optionally incorporating prompts (taxonomy), to generate GO term predictions (MFO, BPO and CCO) with associated probabilities. The protein domain dataset from the InterPRO database and reviewed proteins from the UniProt database are utilized to enhance prediction performance.

Protein Representation learning

In the representation learning stage, a raw amino acid sequence is first passed through ESM2 (3B), producing 2560-dimensional protein embeddings for each amino acid. Simultaneously, a set number of learnable query embeddings are generated as input to the Q-former along with the sequence embeddings. The queries will interact with frozen

protein features through cross-attention structure and interact with input protein function text through self-attention structure. By default, the number of queries is set to 32, its size (32 × 768) is significantly smaller than the size of protein embeddings (sequence length × 2560) generated by ESM2 (3B). This approach enables the model to learn more compact representations while decreasing the total number of parameters.

Protein function generation learning

In the generation stage, we use LLM to generate functional labels such as GO terms. The GO file is of version 1.4 (http://purl.obolibrary.org/obo/go/go-basic.obo), ensuring that the graph is acyclic and allowing for the propagation of annotations up the graph. Specially, LLM processes learned queries from stage 1, along with an additional taxonomy prompt, and then generates text based on this information for molecular function, biological process, and cellular component. Note that the learned queries are preceded by a full connected network, and the taxonomy prompt can also be null.

In terms of the LLM option, instead of OPT-2.7B used in BLIP2, we choose Mistral-7B, a widely used open-source large language model developed by Mistral AI, as a core component of FAPM, due to its superior performance and better development ecosystem. Although Mistral-7B has over twice the number of parameters than OPT-2.7B, the frozen nature of these parameters does not impose a significant burden on the training and inference of the model.

We have included an additional input prompt, not primarily intended to enhance the model's prediction accuracy. For instance, certain prompt categories (e.g., taxonomy) may not substantially improve model performance. Our goal is to introduce interactivity and flexibility to the model. In practical prediction scenarios, proteins are often accompanied by supplementary information, which we aim for the model to engage with and comprehend through prompts, resulting in more precise content. For instance, the inclusion of basic taxonomy information prompts can lead to more reasonable output content and reduce the likelihood of conflicting information.

Pre-Training Objectives

Inspired by BLIP2, we jointly optimize three pre-training objectives that share the same input format and model parameters.

Protein-Text Contrastive Learning (PTC) aims to align the feature space of the protein encoder and the text decoder, encouraging parallel protein-text pairs have higher similarity scores. This objective has been demonstrated to be effective in ALBEF[32] in image-text learning task.

Language Modeling Loss (LM) assess the efficacy of the language model to generate function text given a protein sequence. This metric quantifies the model's proficiency in forecasting the likelihood of subsequent tokens in a sequence, based on the discrepancy between the predicted probabilities and the actual token occurrences.

Protein-Text Matching (PTM) predicts whether a pair of protein and text is positive (matched) or negative (not matched). We follow the negative sampling strategy[32], where negative pairs with higher contrastive similarity in a batch have a higher chance to be sampled. It aims to learn protein-text multimodal representation for protein-text pair that share similar semantics but differ in fine-grained details.

Datasets

UniProtKB/Swiss-Prot Dataset

We downloaded the Swiss-Prot version 2023_04[4], released on October 25, 2023, from the FTP site (https://ftp.uniprot.org/pub/databases/uniprot/previous_releases/), which is a comprehensive resource of protein sequence and annotation information. We filtered sequences shorter than 10 or longer than 1024, then we divided these protein sequences into two parts, the portion automatically annotated was utilized as pre-training samples along with the domain-scale sequences, while the remaining part was used for fine-tuning. Experimental functional annotated proteins can be filtered with evidence codes EXP, IDA, IPI, IMP, IGI, IEP, TAS, IC, HTP, HDA, HMP, HGI and HEP, which contains 71239 reviewed and manually annotated proteins. To fine-tune our model on this dataset and guarantee effective generalization to new proteins, we partitioned the proteins into training, validation, and testing sets. We employed DIAMOND[33] to compute sequence similarity, grouping the proteins based on their similarity before executing a random split. Proteins with sequence identity exceeding 50% were grouped together, with 90% of the groups allocated for training, 5% for validation, and 5% for testing. Table 1 shows the number of samples under the sub-ontologies of GO in the pre-training and fine-tuning phase, as well as the number of GO terms.
To improve the model's ability to learn domain-scale information from sequence, we pre-train the model using domain data generated by InterPro. We utilized data version 95.0 (2023-07), available for download from the InterPro FTP site (https://ftp.ebi.ac.uk/pub/databases/interpro/). However, as these domain sequences are annotated with domain number (such as IPR000001), we utilize the undergoing mapping provided by InterPro to assign GO terms to these sequences.

Stage		MFO	BPO	CCO
Pre-training	train	2173147	1631712	898685
Fine-tuning	train	34589	46422	45298
	val	1818	2454	2442
	test	1858	2417	2388
	terms	8242	21091	2985

Table 1 Summary of pre-training and fine-tuning dataset.

Instruction dataset

We use the protein-oriented instructions introduced by Mol-Instructions (https://huggingface.co/datasets/zjunlp/Mol-Instructions), which covers 505K instructions spanning five categories of tasks. These tasks aim to predict the structure, function, and activity of proteins, and facilitate protein design based on textual directives. In this study, we exclude protein design task due to model limitation. Performance evaluation is conducted using the ROUGE-L[34] metric on all test samples to measure the quality of the model's output against reference answers.

Competing methods

A variety of compared methods are summarized in Table 2, from simplest naïve methods to newest state-of-the-art method DeepGO-SE[35]. AI-based methods mainly rely on protein features or protein sequences as inputs, supplemented with information like PPI or semantic sometimes.

Naive baseline

In a straightforward scenario, we estimate the GO probability of each protein based on the frequency of GO occurrences in the data. This basic approach offers advantages in predicting high-frequency GO tags due to the imbalanced nature of GO category annotations and can serve as a baseline called "naive" method[36]. The prediction scores for GO class f in protein p are computed as:

$$S\left(p\,,f\right) = \frac{N_f}{N_{total}}$$

Where N_f is the number of training proteins annotated by GO class f and N_{total} is a total number of training proteins.

Other methods

- InterPRO is a resource that provides functional analysis of protein sequences by classifying them into families and predicting the presence of domains and important sites through collaboration with multiple member databases (referred to as member databases). Additionally, InterPro provides detailed functional annotations as well as adding relevant GO terms that enable automatic annotation of millions of GO terms across the protein sequence databases.
- DeepGOZero integrates protein function prediction with ELEmbeddings[37], a model-theoretic approach for embedding ontologies into a distributed space. It uses InterPro domain annotations as input and employs two layers of MLPBlock to generate embeddings, then the model learns the embedding space for GO classes using ELEmbeddings' loss functions.
- DeepGraphGO is a multi-species graph neural network-based method for protein function prediction, designed to leverage sequence features (InterPRO domain annotations) and high-order protein network information in an end-to-end manner.
- ProteInfer predicts protein function directly from unaligned amino acid sequences using deep convolutional neural networks. This method could predict Enzyme Commission (EC) numbers and Gene Ontology (GO) terms, while also placing amino acid sequences into a functional space to assist in downstream analysis and interpretation.
- ProtNLM aimed to develop a model that connects amino acid sequences with natural language descriptions of their functional properties. This model was trained to predict free text captions describing these properties. Expert curators evaluated model predictions, leading to performance enhancements. The refined model was then used to predict protein names for around 49 million unknown proteins, now included in the UniProt database.

- DeepFRI[38] is a Graph Convolutional Network that predicts protein functions by utilizing sequence features from a protein language model and structures. DeepFRI uses LSTM[39] language model to extract residue-level features of PDB[40] sequence and GCN[41] with three graph convolutional layers to learn complex structure–function relationships.
- AnnoPRO[42] incorporates sequence-based multi-scale protein representation, dual-path protein encoding through pre-training, and function annotation using long short-term memory-based decoding. It aims to address the challenge of long-tail problem[43].
- DeepGO-SE predicts GO functions from protein sequences. It integrates a pre-trained protein language model with a neuro-symbolic model that leverages GO axioms to perform protein function prediction through approximate semantic entailment.

Methods	Input
Naive	Statistic information
InterPRO	Homology information
DeepGraphGO	Protein features and PPI information
DeepGOZero	Protein features and semantic information
DeepGO-SE	Protein features and semantic information
DeepGOCNN	Sequence
ProteInfer	Sequence
ProtNLM	Sequence
AnnoPRO	Sequence
DeepFRI	Sequence and structure
Mol-Instructions	Sequence

Table 2 Summary of comparable methods.

Experimental settings

We train and evaluate models for each of the sub-ontologies of GO separately. Initially, our methodology employs millions of domain level sequences and several hundred thousand automatically annotated full-chain level sequences in pre-training phase. This phase encompasses a two-stage training process: a 5-epoch protein-text alignment training, followed by a 5-epoch training focused on generating protein functional text outputs. Subsequently, model selection is guided by the validation set's performance metrics. Building upon the pre-trained model, we further refine its performance through a 5-epoch fine-tuning process, utilizing manually annotated data and adopting a lower learning rate of 1e-5.

Evaluation metrics

To summarize model performance as a single scalar, we compute the F_{max}, AUPR and WF_{max}. F_{max}, the maximum F1 score (the geometric mean of precision and recall) across all thresholds, is protein-centric, which has been used in CAFA as the main evaluation metric[44]. AUPR, the area under the precision-recall curve, is pair-centric and widely

used for performance evaluation of multi-label classification including automated function prediction[11, 45]. WF_{max} is weighted version of F_{max}, and the weight of GO annotation is calculated as information accretion (IA)[46].

Results

We evaluate FAPM's overall performance through two comparisons: (1) against other language models for multi-task prediction in the Instruction dataset (see “Datasets” section), and (2) with other popular methods, including InterPRO, DeepGOCNN, DeepGraphGO, DeepGOZero, DeepGO-SE, ProteInfer, ProtNLM, DeepFRI and AnnoPRO, on functional annotation tasks. These competing methods have been briefly described in the “Competing methods” section. Unfortunately, limitations in model open-source availability and training resources prevented us from reproducing some methods or making large-scale predictions, leading to the unavailability of test results on the Swiss-Prot test set. we conducted testing of these methods using a collection of bacteriophage and their bacterial host proteins. Since the experimental annotations for these bacteriophage/bacteria proteins have not been publicly updated, ensuring data integrity, a fair comparison could be conducted.

Comparative Performance Against Other Language Models

Since the FAPM framework incorporates a language model, a straightforward idea is to compare FAPM with other language models to evaluate its performance in protein prediction tasks. We leverage the protein-oriented instruction dataset along with its train/test split to train our model from the beginning. Figure 2 (A) demonstrates the rouge-L metrics of eight methods across four tasks: protein function, functional description, catalytic activity and domain/motif prediction. The protein instance "UMP-CMP kinase" in Figure 2 (B) highlights that FAPM exhibits fewer errors in the generation, better alignment with the ground truth in functional descriptions, and successfully predicts functions ('phosphorylation' and 'nucleus') that the second-best method (Mol-Instructions) failed to capture.

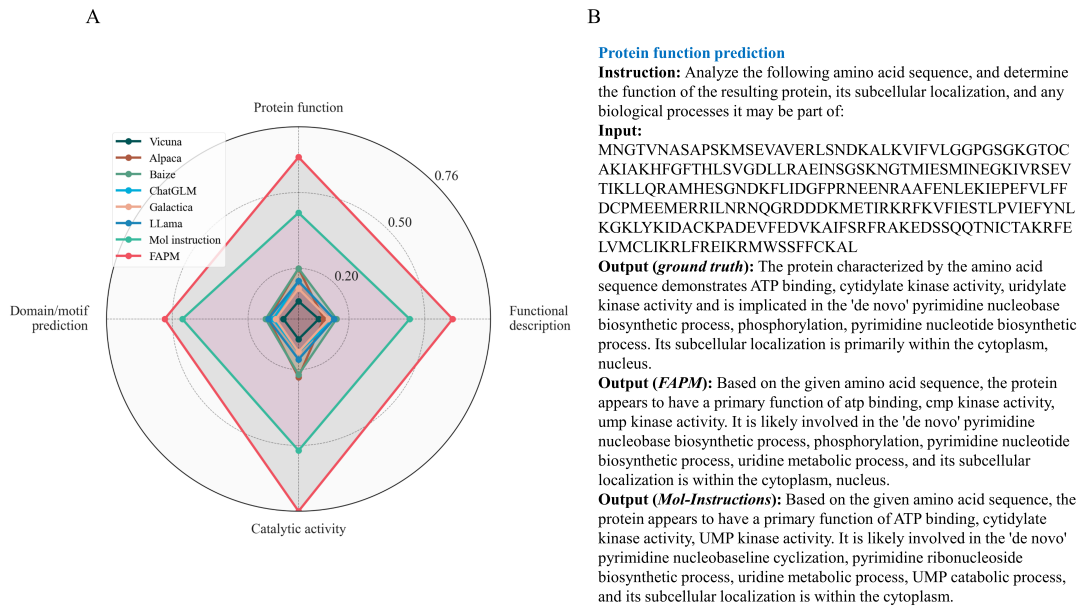

Figure 2 (A) The performance comparison on protein understanding tasks. (B) Examples of predicting protein function using instructions.

Comparison with competing methods on the Swiss-Prot dataset

As shown in table 3, we made a comparison of FAPM with competing methods: Naive, DeepGOCNN, DeepGraphGO and DeepGOZero. It's important to note that due to open -source or computing resource limitations, ProteInfer, DeepFRI and AnnoPRO cannot be retrained, while ProtNLM is primarily used for predicting protein names and does not easily compute evaluation metrics. As a result, we did not compare these methods in this section. According to table 3, we have five findings:

(1) $FAPM_{ESM2\text{-}3B+prompt}$ has the best performance of F_{max}, WF_{max}, AUPR and S_{min} in MFO, BPO and CCO. For example, it achieved the highest F_{max} of 0.586 in MFO, which was 5.2% improvement over DeepGO-SE (0.534).

(2) $FAPM_{ESM2\text{-}3B+prompt}$ achieves relatively small advantage across some metrics in the BPO domain, with 2.4% improvement over DeepGraphGO in AUPR. The explanation can be inferred from Figure 2(A): when the recall value is approximately between 0 and 0.3, the precision value of DeepGraphGO is significantly higher than that of other methods, leading to its higher AUPR value. This could be attributed to its effective utilization of PPI network information[13].

(3) Compared to the Naive method, other models achieve a more significant advantage in the WF_{max} metric other than F_{max}, as improved predictive performance for rare annotations leads to higher WF_{max} scores. For example, in CCO, the F_{max} of DeepGOZero (0.593) is close to Naive (0.565), however, the WF_{max} of DeepGOZero (0.474) is far more than Naive (0.351). As we can see, FAPM achieved both advantages in F_{max} and WF_{max}, showing its proficiency in predicting both common and rare GO annotations.

(4) The sequence-based deep learning method, DeepGOCNN, performed poorly across all three GO domains. This suggests that using a basic convolutional neural network to encode protein sequences may struggle to capture the most relevant information for protein function prediction.

(5) Compared to $FAPM_{ESM2\text{-}650M}$, $FAPM_{ESM2\text{-}650M+prompt}$ performs better, especially in the bp and cc domain. We guess that the species information in the prompt is related to special functions. The case of homologues proteins with distinct functions in Figure 3 supports our suspicions. Inputting taxonomy (such as "Homo") in the form of a prompt as exterior supplementary to the language side of FAPM, we monitored the changes of the model's generated abilities. Encouragingly, we noticed improvements in the generated content, such as avoiding some basic errors and enhancing the quality of our predictions. For instance, considering a pair of homologous proteins, TRPV1_HUMAN and A0A452GR26_9SAUR, categorized as Homo and Phasianus (a type of turkey) in the taxonomic hierarchy. They exhibit a high degree of similarity, with a sequence similarity of 70.76% and a structural RMSD value down to 1.593. However, a crucial difference lies in their abilities to function as capsaicin receptors (TRPV1), which is a heat-activated cation channel that is modulated by inflammatory agents and contributes to acute and persistent pain[47]. As we know, the seeds of Capsicum plants are dispersed predominantly by birds. In birds, the TRPV1 channel does not respond to capsaicin or related chemicals but mammalian TRPV1 is very sensitive to it, in contrast, mammalian TRPV1 channels are responsive to capsaicin due to their functional diversity and selective sensitivity, resulting in a spicy taste sensation. The function of TRPV1 is associated with "behavioral response to pain" (GO:0048266) in the GO system. With the species information, our model correctly annotated TRPV1_HUMAN with GO:0048266 while not for A0A452GR26_9SAUR, which is consistent with the aforementioned knowledge.

The precision-recall curves of six methods on the test set of UniProtKB-SwissProt dataset are plotted in Figure 4(A). It is worth noting that DeepGOCNN did not outperform the Naive method by much, while other deep learning methods significantly surpassed the Naive method, especially in the MFO domain. Interestingly, in the BPO domain, if one aims for high precision in prediction results (at the cost of low recall), then DeepGraphGO is preferable. However, overall, FAPM tends to perform better. In the CCO domain, the precision-recall curve of the Naive method crosses with that of other methods, indicating its prediction advantage in some cases (when recall is around or below 0.5).

Figure 4(B) illustrates the relationship between the frequency of GO term occurrences and the predictive performance of the models. We counted the occurrences of each GO term on the training set and grouped them into five categories: 0-20, 20-50, 50-200, 200-1000, and 1000+. The average F1 score for the predicted GO terms in each group was then calculated to compare the performance of four models: DeepGraphGO, DeepGOZero, DeepGO-SE and FAPM. When predicting high-frequency GO terms (1000+), the three methods show similar performance, with FAPM slightly ahead. However, for GO terms occurring less than 1000 times, FAPM demonstrates a more significant advantage, especially in BPO and CCO categories, highlighting FAPM's strength in predicting rare GO terms.

MFO	F_{max} (↑)	WF_{max} (↑)	AUPR (↑)
Naive	0.292	0.18	0.148
DeepGOCNN	0.306	0.211	0.187
DeepGraphGO	0.501	0.441	0.382
DeepGOZero	0.523	0.474	0.416
DeepGO-SE	0.534	0.495	0.428
$FAPM_{ESM2\text{-}650M}$	0.569	0.536	0.45
$FAPM_{ESM2\text{-}650M+prompt}$	0.574	0.539	0.45
$FAPM_{ESM2\text{-}3B+prompt}$	**0.586**	**0.556**	**0.462**

BPO	F_{max} (↑)	WF_{max} (↑)	AUPR (↑)
Naive	0.322	0.2	0.224
DeepGOCNN	0.339	0.216	0.236
DeepGraphGO	0.438	0.336	0.414
DeepGOZero	0.415	0.31	0.318
DeepGO-SE	0.437	0.336	0.353
$FAPM_{ESM2\text{-}650M}$	0.507	0.425	0.414
$FAPM_{ESM2\text{-}650M+prompt}$	0.509	0.431	0.419
$FAPM_{ESM2\text{-}3B+prompt}$	**0.527**	**0.453**	**0.438**
CCO	F_{max} (↑)	WF_{max} (↑)	AUPR (↑)
Naive	0.565	0.351	0.437
DeepGOCNN	0.586	0.448	0.457
DeepGraphGO	0.593	0.466	0.459
DeepGOZero	0.593	0.474	0.454
DeepGO-SE	0.651	0.571	0.519
$FAPM_{ESM2\text{-}650M}$	0.673	0.609	0.533
$FAPM_{ESM2\text{-}650M+prompt}$	0.696	0.643	**0.547**
$FAPM_{ESM2\text{-}3B+prompt}$	**0.703**	**0.652**	0.544

Table 3 Prediction results for MFO, BPO and CCO on the test set of UniProtKB Swiss-Prot dataset.

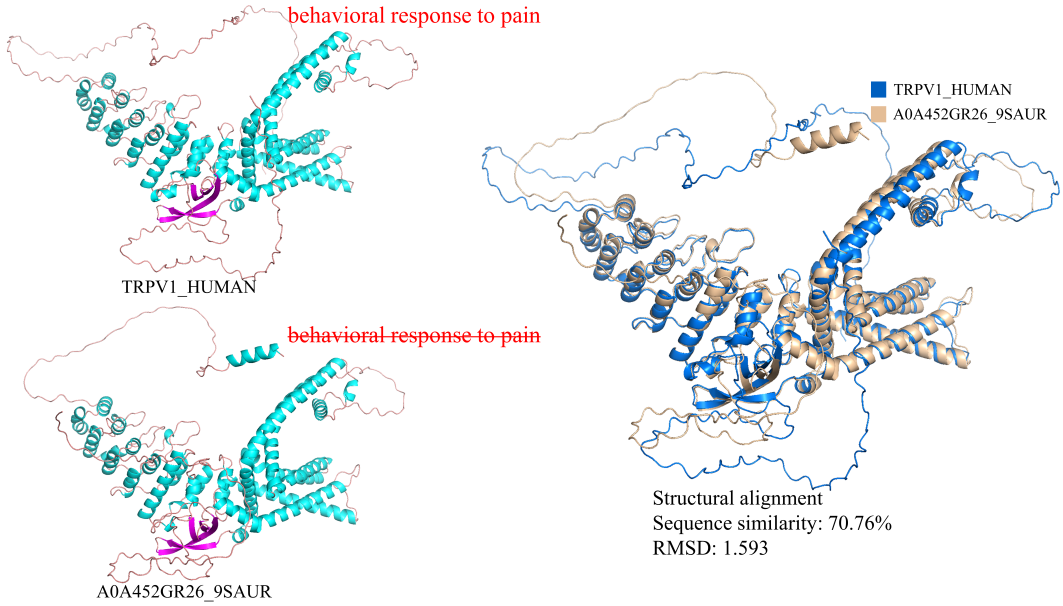

Figure 3 Case of functional prediction of homologous proteins from different species.

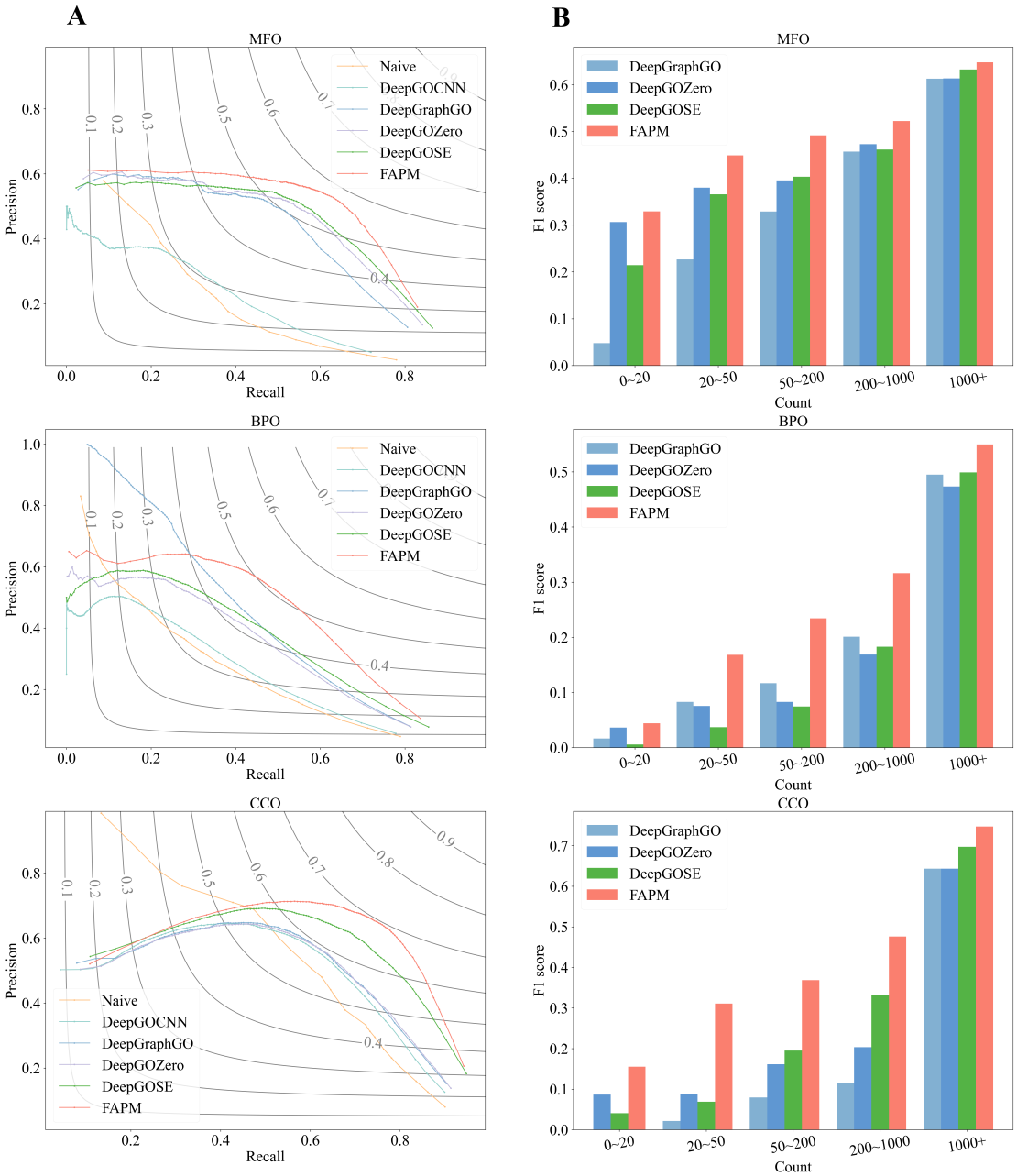

Figure 4 (**A**) Recall-precision curve of various methods for Molecular Function, Biological Process and Cellular Component prediction on the test set of UniProtKB Swiss-Prot dataset. (**B**) The relationship between the frequency of Gene Ontology (GO) labels on the training set and the F1 scores of predictions on the test set of UniProtKB Swiss-Prot dataset.

The explainability of multi-modal representation

FAPM achieves better quality protein representations in terms of properties compared to pretrained protein language model such as ESM2. Figure 5 visualizes the 2D projections of proteins with specific functions from the test set of the Swiss-Prot dataset using UMAP[48]. The figure consists of three parts (A, B, and C), each comparing FAPM to ESM2. Part A focuses on most common functions (proteins with multi

functions removed), while parts B and C show proteins annotated with similar and contrasting functions, respectively. In all three scenarios, the representations derived from ESM2 exhibit weak clustering, whereas FAPM yields significantly distinct and intriguing results. Specifically, part B (right) presents two closely positioned clusters, and part C (right) presents two relatively wider separated clusters, demonstrating that our model effectively captures the functional relationships of protein sequences. Overall, the representations learned by FAPM not only outperform those of ESM2 but also align with intuitive expectations about cluster distances, suggesting that our model successfully incorporates semantic information from property descriptions.

Figure 5 The comparisons of protein representations with specific GO functions between ESM2 and FAPM. All samples are from test set of Swiss-Prot dataset. (A) proteins annotated with most common functions. (B) proteins annotated with DNA/RNA binding (similar functions). (C) proteins annotated with kinase/phosphatase activity (contrast functions).

Bacteriophage protein prediction cases

Predicting the functions of proteins without close homologs is challenging for both homology search and AI-based methods. To fully exploit the potential of our model, we chose bacteria and bacteriophage (phage) proteins to experimentally validate our predictions. Bacteria are the leading cause of pathogenic diseases and commensal microorganisms of human. Our understandings on bacteria are largely based on model bacteria, yet Escherichia coli (E. coli)[49] and Bacillus subtilis (B. subtilis)[50], representing Gram-negatives and Gram-positives, still requires vigorous further investigation as more than half of their genes still need functional assignment. Phages have an estimated population of 10^{31}[51], more than all other lives combined on earth. The enormous population and variety of phage make their gene products largest reservoir of proteins with unknown functions, which are impossible to be annotated manually based on current experimental skill sets. We selected a wide range of bacteria and phage proteins, including those from E. coli, B. subtilis, phages for common bacteria, and rare RNA phage. Out of the 29 proteins (24 from phage and 5 from bacteria) selected, 3 recently assigned, and 5 are experimental validated but not described in the database uniport (Supplementary Table S1).

we evaluated FAPM against other six methods in predicting experimental GO annotations of bacteriophage proteins that mostly have no homology information available. Our analysis steps are as follows:

(1) For these proteins, we conducted a search in the InterPRO database to obtain homology information, and also assessed their similarity with sequences in Swiss-Prot. This analysis aids in understanding the predictability of the sequence. For example, in InterPRO we can search that Gp57B has been annotated IPR009097 (Cyclic phosphodiesterase), which is defined as β-barrel domain consisting of a duplication of a β/α/β/α/β motif found in cyclic phosphodiesterase (CNPase)[52-54]. And this definition can be considered as GO:0016787 (hydrolase activity). For similarity-based algorithms and AI algorithms, proteins with identifiable homology information, like Gp57B, are relatively easier to predict. However, there are many other proteins lacking homology information, such as Gp27, present a greater challenge for function prediction.

(2) We employ various methods to predict protein functions, which can be classified into three categories. The first category consists of a single method that utilizes the homology search results from InterPRO for prediction, serving as our baseline method. The second category includes DeepGOZero, which are based on sequence annotation. We excluded DeepGOCNN due to its poor performance (shown in 3.1) and DeepGraphGO because of the lack of protein interaction data. The third category focuses on purely sequence or structure-based methods, such as ProteInfer, ProtNLM, DeepFRI and AnnoPRO, all of which are trained on large scale datasets.

(3) It's important to note that, DeepGOZero, DeepGO-SE, ProtNLM, and AnnoPRO provide probabilities for nearly all GO labels, prompting us to select the top 10 predictions based on their scores. ProteInfer and DeepFRI only output the most likely predictions, so we did not further filter their predictions, which could be either more or fewer than 10.

We tabulated the occurrences of experimental annotations on the bacteriophage proteins, with "DNA binding" being the most frequent annotation, appearing 11 times. "RNA binding" appeared twice, while "Nuclease activity," "Catalase

activity," and "Peptidase inhibitor activity" each appeared once. The recall, precision, and f1 score for these annotations across various methods are computed and presented in Table 4. However, due to the lack of predictive results of some methods for some GO labels, part of results is therefore not given.

For prediction of DNA binding, $FAPM_{ESM2\text{-}3B+prompt}$ achieved F1 score of 88%, followed by DeepGO-SE with 69%. Notice that although both ProtNLM and DeepFRI have 100% precision, their recall values are too low, resulting in lower F1 scores. In the case of RNA binding, FAPM gave a successful prediction for Gp49, and none of the other methods gave a correct prediction for this annotation. The remaining 3 annotations ("Nuclease activity", "Catalase activity" and "Peptidase inhibitor activity") were successfully predicted by FAPM, whereas ProtNLM, AnnoPRO and DeepGO-SE made accurate predictions for two of them. The test results from these cases demonstrate that FAPM exhibits consistent and outstanding performance in annotation prediction, even in some challenging cases.

GO annotations	Methods	Recall	Precision	F1 score
DNA binding	InterPRO	/	/	/
	DeepGOZero	/	/	/
	ProteInfer	/	/	/
	ProtNLM	27%	100%	43%
	DeepFRI	18%	100%	31%
	AnnoPRO	36%	67%	47%
	DeepGO-SE	82%	69%	69%
	$FAPM_{ESM2\text{-}650M+prompt}$	82%	75%	78%
	$FAPM_{ESM2\text{-}3B+prompt}$	100%	79%	**88%**
RNA binding	InterPRO	/	/	/
	DeepGOZero	/	/	/
	ProteInfer	0%	0%	0%
	ProtNLM	0%	0%	0%
	DeepFRI	0%	0%	0%
	AnnoPRO	0%	0%	0%
	DeepGO-SE	0%	0%	0%
	$FAPM_{ESM2\text{-}650M+prompt}$	50%	33%	**40%**
	$FAPM_{ESM2\text{-}3B+prompt}$	50%	25%	33%
Others (Nuclease activity, Catalase activity, Peptidase inhibitor activity)	InterPRO	100%	100%	**100%**
	DeepGOZero	33%	100%	50%
	ProteInfer	33%	100%	50%
	ProtNLM	67%	100%	80%
	DeepFRI	0%	0%	0%
	AnnoPRO	67%	100%	80%
	DeepGO-SE	67%	100%	80%
	$FAPM_{ESM2\text{-}650M+prompt}$	100%	100%	**100%**
	$FAPM_{ESM2\text{-}3B+prompt}$	100%	100%	**100%**

Table 4 Recall, Precision and F1 Score of different methods for predicting GO terms of different bacteriophage proteins.

Discussion

FAPM utilizes a protein pre-training model, ESM2, to extract sequence information. And by aligning the sequence information with the natural language information, FAPM can use language model Mistral-7b to generate function text. Therefore, FAPM is a sequence-based protein function prediction model, FAPM, based on contrastive learning. In general, FAPM has three advancements in field:

(1) Protein sequence is the only required input, and the output is easy to interpret due to the utilization of a language model.
(2) Millions of protein domain data and hundreds of thousands of manually labeled data were used for model training. Our model demonstrates superior performance when compared to the alternative methods. While predicting the unannotated bacteriophage/bacteria proteins, our model exhibits better accuracy and generalization capabilities in predicting non-homologous proteins.
(3) The model can interact through prompts. The taxonomy used as a prompt in this study, significantly enhances the coherence of the output. Theoretically, more diverse prompts could be introduced to meet the demands of complex real-world scenarios.

In summary, FAPM holds significant potential for further development. For comparison and testing purposes, we have temporarily restricted the text generated by the model to the GO labelling system. However, the model is capable of interacting with prompts, and with appropriate improvements, it may make richer and more flexible predictions, including protein modification and generation. Moreover, enhancing the overall performance can be conveniently achieved by utilizing more powerful protein models and language models.

Data and code availability

Code:
https://github.com/xiangwenkai/FAPM
Data:
https://ftp.uniprot.org/pub/databases/uniprot/previous_releases/release-2023_04/knowledgebase/

Funding

This work was supported by Shanghai Rising-Star Program (23QD1400600), National Natural Science Foundation of China (82204278 to XTL, T2225002 to MYZ), and National Key Research and Development Program of China (2022YFC3400504 to

MYZ).

Reference

1. Jumper, J., et al., Highly accurate protein structure prediction with AlphaFold. Nature, 2021. **596**(7873): p. 583-589.
2. Varadi, M., et al., AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. Nucleic acids research, 2022. **50**(D1): p. D439-D444.
3. Boeckmann, B., et al., The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic acids research, 2003. **31**(1): p. 365-370.
4. UniProt: the universal protein knowledgebase in 2023. Nucleic Acids Research, 2023. **51**(D1): p. D523-D531.
5. Ashburner, M., et al., Gene ontology: tool for the unification of biology. Nature genetics, 2000. **25**(1): p. 25-29.
6. Altschul, S.F., et al., Basic local alignment search tool. Journal of molecular biology, 1990. **215**(3): p. 403-410.
7. Bairoch, A., PROSITE: a dictionary of sites and patterns in proteins. Nucleic acids research, 1991. **19**(Suppl): p. 2241.
8. Krogh, A., et al., Hidden Markov models in computational biology: Applications to protein modeling. Journal of molecular biology, 1994. **235**(5): p. 1501-1531.
9. Eddy, S.R., Profile hidden Markov models. Bioinformatics (Oxford, England), 1998. **14**(9): p. 755-763.
10. Blum, M., et al., The InterPro protein families and domains database: 20 years on. Nucleic acids research, 2021. **49**(D1): p. D344-D354.
11. Kulmanov, M. and R. Hoehndorf, DeepGOPlus: improved protein function prediction from sequence. Bioinformatics, 2020. **36**(2): p. 422-429.
12. Kulmanov, M. and R. Hoehndorf, DeepGOZero: improving protein function prediction from sequence and zero-shot learning based on ontology axioms. Bioinformatics, 2022. **38**(Supplement_1): p. i238-i245.
13. You, R., et al., DeepGraphGO: graph neural network for large-scale, multispecies protein function prediction. Bioinformatics, 2021. **37**(Supplement_1): p. i262-i271.
14. Chiang, W.-L., et al. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. in Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. 2019.
15. Nooren, I.M. and J.M. Thornton, Diversity of protein–protein interactions. The EMBO journal, 2003. **22**(14): p. 3486-3492.
16. Szklarczyk, D., et al., STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. Nucleic acids research, 2019. **47**(D1): p. D607-D613.
17. Krizhevsky, A., I. Sutskever, and G.E. Hinton, Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012. **25**.
18. Sanderson, T., et al., ProteInfer, deep neural networks for protein functional inference. Elife, 2023. **12**: p. e80942.
19. Vaswani, A., et al., Attention is all you need. Advances in neural information processing systems, 2017. **30**.
20. Zhang, H., et al., A survey of controllable text generation using transformer-based pre-trained language models. ACM Computing Surveys, 2023. **56**(3): p. 1-37.
21. Liu, Y., et al., A survey of visual transformers. IEEE Transactions on Neural Networks and Learning Systems, 2023.
22. Brandes, N., et al., ProteinBERT: a universal deep-learning model of protein sequence and function. Bioinformatics, 2022. **38**(8): p. 2102-2110.
23. Kabir, A. and A. Shehu, GOProFormer: A Multi-Modal Transformer Method for Gene Ontology Protein Function Prediction. Biomolecules, 2022. **12**(11): p. 1709.
24. Lin, Z., et al., Language models of protein sequences at the scale of evolution enable accurate structure prediction. BioRxiv, 2022. **2022**: p. 500902.
25. Clifford, J.N., et al., BepiPred-3.0: Improved B-cell epitope prediction using protein language models. Protein Science, 2022. **31**(12): p. e4497.
26. Qiu, J., et al., Large ai models in health informatics: Applications, challenges, and the future. IEEE Journal of Biomedical and Health Informatics, 2023.

27. Ahdritz, G., et al., OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization. bioRxiv, 2022: p. 2022.11. 20.517210.
28. Taylor, R., et al., Galactica: A large language model for science. arXiv preprint arXiv:2211.09085, 2022.
29. Fang, Y., et al., Mol-instructions: A large-scale biomolecular instruction dataset for large language models. arXiv preprint arXiv:2306.08018, 2023.
30. Jiang, A.Q., et al., Mistral 7B. arXiv preprint arXiv:2310.06825, 2023.
31. Devlin, J., et al., Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
32. Li, J., et al., Align before fuse: Vision and language representation learning with momentum distillation. Advances in neural information processing systems, 2021. **34**: p. 9694-9705.
33. Buchfink, B., C. Xie, and D.H. Huson, Fast and sensitive protein alignment using DIAMOND. Nature methods, 2015. **12**(1): p. 59-60.
34. Lin, C.-Y. Rouge: A package for automatic evaluation of summaries. in Text summarization branches out. 2004.
35. Kulmanov, M., et al., Deepgo-se: Protein function prediction as approximate semantic entailment. bioRxiv, 2023: p. 2023.09. 26.559473.
36. Radivojac, P., et al., A large-scale evaluation of computational protein function prediction. Nature methods, 2013. **10**(3): p. 221-227.
37. Kulmanov, M., et al., El embeddings: Geometric construction of models for the description logic el++. arXiv preprint arXiv:1902.10499, 2019.
38. Gligorijevi¿, V., et al., Structure-based protein function prediction using graph convolutional networks. Nature communications, 2021. **12**(1): p. 3168.
39. Shi, X., et al., Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Advances in neural information processing systems, 2015. **28**.
40. Burley, S.K., et al., Protein Data Bank (PDB): the single global macromolecular structure archive. Protein crystallography: methods and protocols, 2017: p. 627-641.
41. Kipf, T.N. and M. Welling, Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907, 2016.
42. Zheng, L., et al., AnnoPRO: a strategy for protein function annotation based on multi-scale protein representation and a hybrid deep learning of dual-path encoding. Genome Biology, 2024. **25**(1): p. 41.
43. Unsal, S., et al., Learning functional properties of proteins with language models. Nature Machine Intelligence, 2022. **4**(3): p. 227-245.
44. Jiang, Y., et al., An expanded evaluation of protein function prediction methods shows an improvement in accuracy. Genome biology, 2016. **17**(1): p. 1-19.
45. Kulmanov, M., M.A. Khan, and R. Hoehndorf, DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. Bioinformatics, 2018. **34**(4): p. 660-668.
46. Clark, W.T. and P. Radivojac, Information-theoretic evaluation of predicted ontological annotations. Bioinformatics, 2013. **29**(13): p. i53-i61.
47. Basbaum, A.I., et al., Cellular and molecular mechanisms of pain. Cell, 2009. **139**(2): p. 267-284.
48. McInnes, L., J. Healy, and J. Melville, Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426, 2018.
49. Nataro, J.P. and J.B. Kaper, Diarrheagenic escherichia coli. Clinical microbiology reviews, 1998. **11**(1): p. 142-201.
50. Kovács, Á.T., Bacillus subtilis. Trends in microbiology, 2019. **27**(8): p. 724-725.
51. Weitz, J.S., et al., Phage–bacteria infection networks. Trends in microbiology, 2013. **21**(2): p. 82-91.
52. Kozlov, G., et al., Solution structure of the catalytic domain of RICH protein from goldfish. The FEBS Journal, 2007. **274**(6): p. 1600-1609.
53. Remus, B.S., A. Jacewicz, and S. Shuman, Structure and mechanism of E. coli RNA 2′, 3′-cyclic phosphodiesterase. Rna, 2014. **20**(11): p. 1697-1705.
54. Hofmann, A., et al., Crystal Structures of the Semireduced and Inhibitor-bound Forms of Cyclic Nucleotide Phosphodiesterase from Arabidopsis thaliana* 210. Journal of Biological Chemistry, 2002. **277**(2): p. 1419-1425.

Supplementary

Name	location	Functional labels
Gp41	cytosol	DNA binding; protein-protein interaction
Gp44	cytosol	DNA binding
Gp45	cytosol	DNA binding
Gp46	cytosol	Protein-protein interaction
Gp49	cytosol	RNA binding
Gp60	cytosol	Protein-protein interaction
Gp33	cytosol	Protein-protein interaction
Gp27	cytosol	DNA binding
Gp35.1	cytosol	Toxin activity
RpbA	cytosol	Protein-protein interaction
Exod	cytosol	Dual nuclease activity
Gp57B	cytosol	Catalase activity
Pin	cytosol	Peptidase inhibitor activity; Protein-protein interaction
Mrh	cytosol	DNA binding; Protein-protein interaction
Cef	cytosol	RNA binding; Protein-protein interaction
MsyB	cytosol	Protein-protein interaction, E. coli
YciZ	cytosol	Signal transduction; Protein-protein interaction, B. subtilis
YlaB	inner membrane	Signal transduction; Protein-protein interaction, B. subtilis
Ysdb	inner membrane	Signal transduction; Protein-protein interaction, B. subtilis
RsiW	transmembrane	Signal transduction; Protein-protein interaction, B. subtilis
Gp2	cytosol	Protein-protein interaction
Gp6	cytosol	DNA binding
Gp8	cytosol	DNA binding
Gp12	cytosol	DNA binding
ss1		DNA binding
sm2	cytosol	Protein-protein interaction
sl1	cytosol	Protein-protein interaction
75	cytosol	DNA binding
235	cytosol	DNA binding

Supplementary Table S1 the experimentally verified bacteriophage and bacteria protein that are unannotated in UniProt. Proteins are bacteriophage protein unless listed otherwise.