nature machine intelligence

Article

https://doi.org/10.1038/s42256-024-00795-w

Protein function prediction as approximate semantic entailment

Received: 12 August 2023

Accepted: 10 January 2024

Published online: 14 February 2024

Check for updates

Maxat Kulmanov [1,2,3] ✉, Francisco J. Guzmán-Vega [2,4], Paula Duek Roggli [5,6], Lydie Lane [5,6], Stefan T. Arold [2,4] & Robert Hoehndorf [1,2,3] ✉

The Gene Ontology (GO) is a formal, axiomatic theory with over 100,000 axioms that describe the molecular functions, biological processes and cellular locations of proteins in three subontologies. Predicting the functions of proteins using the GO requires both learning and reasoning capabilities in order to maintain consistency and exploit the background knowledge in the GO. Many methods have been developed to automatically predict protein functions, but effectively exploiting all the axioms in the GO for knowledge-enhanced learning has remained a challenge. We have developed DeepGO-SE, a method that predicts GO functions from protein sequences using a pretrained large language model. DeepGO-SE generates multiple approximate models of GO, and a neural network predicts the truth values of statements about protein functions in these approximate models. We aggregate the truth values over multiple models so that DeepGO-SE approximates semantic entailment when predicting protein functions. We show, using several benchmarks, that the approach effectively exploits background knowledge in the GO and improves protein function prediction compared to state-of-the-art methods.

Protein function prediction is one of the key challenges in modern biology and bioinformatics as it enables better understanding of the roles and interactions of proteins within living systems. Accurate functional descriptions of proteins are necessary for tasks such as identification of drug targets, understanding disease mechanisms and improving biotechnological applications in industry. While predicting protein structures has become increasingly accurate in recent years[1], predicting protein function remains challenging due to the small number of known functions combined with their complexity and interactions.

Functions of proteins are described using the Gene Ontology (GO)[2] which is one of the most successful ontologies in biology. GO includes three subontologies for describing molecular functions (MFO) of a single protein, biological processes (BPO) to which proteins can contribute and cellular components (CCO) where proteins are active. Researchers identify protein functions based on experiments and generate scientific reports which are then taken by database curators and added to knowledge bases. These annotations are generally propagated to homologue proteins. As a result, the UniProtKB/Swiss-Prot database[3] contains manually curated GO annotations for thousands of organisms and more than 550,000 proteins.

Recent protein function prediction methods rely on different sources of information such as sequence, interactions, protein tertiary structure, literature, coexpression, phylogenetic analysis or the information provided in GO[4–20]. The methods may use sequence domain annotations[5,6,8,11,21], directly apply deep convolutional neural networks (CNN)[13] or language models such as long short-term memory neural networks[9] and transformers[14], or use pretrained protein language models[10,15] to represent amino acid sequences. Models may

[1]Computer, Electrical and Mathematical Sciences & Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. [2]Computational Bioscience Research Center, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. [3]SDAIA-KAUST Center of Excellence in Data Science and Artificial Intelligence, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia. [4]Bioscience Program, Biological and Environmental Science and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. [5]CALIPHO group, SIB Swiss Institute of Bioinformatics, CMU, Geneva, Switzerland. [6]Department of Microbiology and Molecular Medicine, Faculty of Medicine, University of Geneva, CMU, Geneva, Switzerland. ✉e-mail: maxat.kulmanov@kaust.edu.sa; robert.hoehndorf@kaust.edu.sa

also incorporate protein–protein interactions through knowledge graph embeddings[12,16], approaches using k nearest neighbours[21] and graph convolutional neural networks[6]. Also, natural language models applied to scientific literature have been successful in automated function prediction[8].

One of the major limitations of many function prediction methods is their reliance on sequence similarity to predict functions. While this approach has been effective when applied to proteins that have similar proteins with well-characterized functions, it can be less reliable for proteins with little or no sequence similarity to known functional domains. Molecular functions arise largely from structure, and proteins with similar structures might have different sequence[22]. Importantly, proteins with similar sequences can have a different set of functions depending on their active sites and the organisms in which they are a part. Consequently, methods that use the same sources of information for all three subontologies of GO are limited; while functions from the MFO subontology can be predicted by a protein sequence or structure, functions from BPO and, to a lesser degree, CCO, inherently rely on multiple proteins being present and interacting in particular ways; therefore, predicting BPO and CCO annotations requires different sources of information than predicting MFO annotations. In general, predicting whether a protein participates in a biological process requires knowledge of an organisms proteome, or at least its annotated genome so that proteins can be predicted; as a result, two proteins may have 100% sequence identity but participate in different processes, depending on the presence or absence of other proteins within the organism's proteome. Protein–protein interaction networks can encode the proteome as well as limit the search space for potential interactions between proteins that give rise to biological processes.

Ontologies are another source of information rarely exploited for predicting protein functions. Ontologies are not simply collections of classes; rather, ontologies are formal theories that specify some aspects of the intended meaning of a class using a logic-based language[23]. The background knowledge that is contained in the axioms of GO can be used by some machine learning models to improve predictions through knowledge-enhanced machine learning[11,12,14,15]. By incorporating the formal axioms into machine learning models, it becomes possible to leverage prior knowledge during the learning or prediction process, put constraints on the parameter search space that can improve the accuracy and efficiency of the learning process and, ultimately, make better predictions[24,25]. While there are different approaches of how formal background knowledge can be incorporated in machine learning methods, approximate entailment aims to explicitly and provably perform 'semantic entailment' as optimization objective, and therefore reproduce many of the formal properties of deductive systems[26]. Only few function prediction methods utilize the formal axioms that are in GO. Hierarchical classification methods for predicting protein functions such as GoStruct2 (ref. 27), DeepGO[12] DeePred[28], SPROF-GO[29] and TALE[14] use subsumption axioms to extract hierarchical relations between classes but ignore other axioms in GO that can be used to reduce the search space and improve predictions.

We have developed DeepGO-SE, a protein function prediction method which predicts functions from protein sequences using a pretrained large protein language model combined with a neuro-symbolic model that performs function prediction as approximate semantic entailment. We use the ESM2 protein language model[30] to generate representations of single proteins. Similar to DeepGOZero[11], we project the ESM2 embeddings into an embedding space (ELEmbeddings) that is generated from the axioms in the GO[31]. ELEmbeddings encode ontology axioms based on geometric shapes and geometric relations, and corresponds to a Σ algebra, or 'world model', in which we can determine whether statements are true or false. In contrast to DeepGOZero, we use these world models to perform 'semantic entailment': statement ϕ is entailed by theory T ($T \models \phi$) if and only if ϕ is true in every world model in which all statements in T are true[32]. While there are, in general, infinitely many such world models for a theory T or a statement ϕ, we learn multiple, but finitely many, such models and generate predictions of functions as 'approximate' semantic entailment where we test for truth in each of the generated world models. Using this form of approximate semantic entailment, we show that the axioms in the extended version of GO enhance the predictions of molecular functions.

Furthermore, we improve predictions for complex biological processes and cellular components by incorporating information about an organism's proteome and interactome in the form of protein–protein interaction networks. We show that, unlike molecular functions, predictions of annotations to biological processes and cellular components can substantially benefit from protein–protein interactions. For biological processes, we found that integrating predicted molecular functions and interactions considerably improves the performance of the predictions; this finding indicates that the prediction of biological process annotations does not require knowledge of specific proteins but only their molecular functions, thereby substantially expanding the generality of our method.

We train and evaluate our model on a dataset with experimental annotations which is split based on sequence similarity to make sure that the evaluations are reported using a test set that does not share similar protein with the training set. We find that methods which rely on sequence similarity perform poorly in this setting, whereas DeepGO-SE significantly improves the prediction performance for all subontologies of GO. For example, DeepGOPlus[13], which predicts functions using both sequence similarity and a convolutional neural network (CNN), can only rely on its CNN and its performance drops on this test set.

Overall, the contributions of our work are as follows:

- We developed a method for knowledge-enhanced machine learning as approximate semantic entailment over multiple generated world models.
- We developed a method for predicting protein functions which improves the prediction performance of subontologies of GO by using knowledge-enhanced learning and a combination of different sources of information.
- We improve the function prediction performance for novel proteins by using sequence features generated by a pretrained protein language model ESM2.

Results

DeepGO semantic entailment

The DeepGO-SE model implements knowledge-enhanced learning by approximating semantic entailment. DeepGO-SE performs knowledge-enhanced learning in three steps. First, we generate an approximate model $\mathcal{I}$ using ELEmbeddings[31] based on the logical theory $\mathcal{O}$ which consists of background knowledge (that is, axioms) in the GO and a set of assertions about proteins (statements of the type 'protein has function C'). Then, we represent proteins by ESM2 (ref. 30) embeddings and use them as instances in the approximate model $\mathcal{I}$ such that the truth of the statement 'protein has function C' is maximized in $\mathcal{I}$ as an optimization objective (that is, $\mathcal{I} \models \phi$ should hold). Finally, we repeat this procedure and generate k approximate models $\mathcal{I}_1, \ldots, \mathcal{I}_k$ of $\mathcal{O}$; entailment is defined as truth in all models ($\mathcal{O} \models \phi$ iff $\mathrm{Mod}(\mathcal{O}) \subseteq \mathrm{Mod}(\{\phi\})$), and the k models are used for approximate entailment. To compute entailments, we aggregate the truth of the statements 'protein has function C' over all generated models. Figure 1 shows this process, and section 'Approximate semantic entailment' provides more details.

UniProtKB/Swiss-Prot dataset evaluation

We evaluate and compare our method with the baseline methods using the UniProtKB/Swiss-Prot dataset split by sequence similarity. We use protein-centric evaluation measures such as maximum F measure (F max), minimum semantic distance (S min) and area under

https://doi.org/10.1038/s42256-024-00795-w

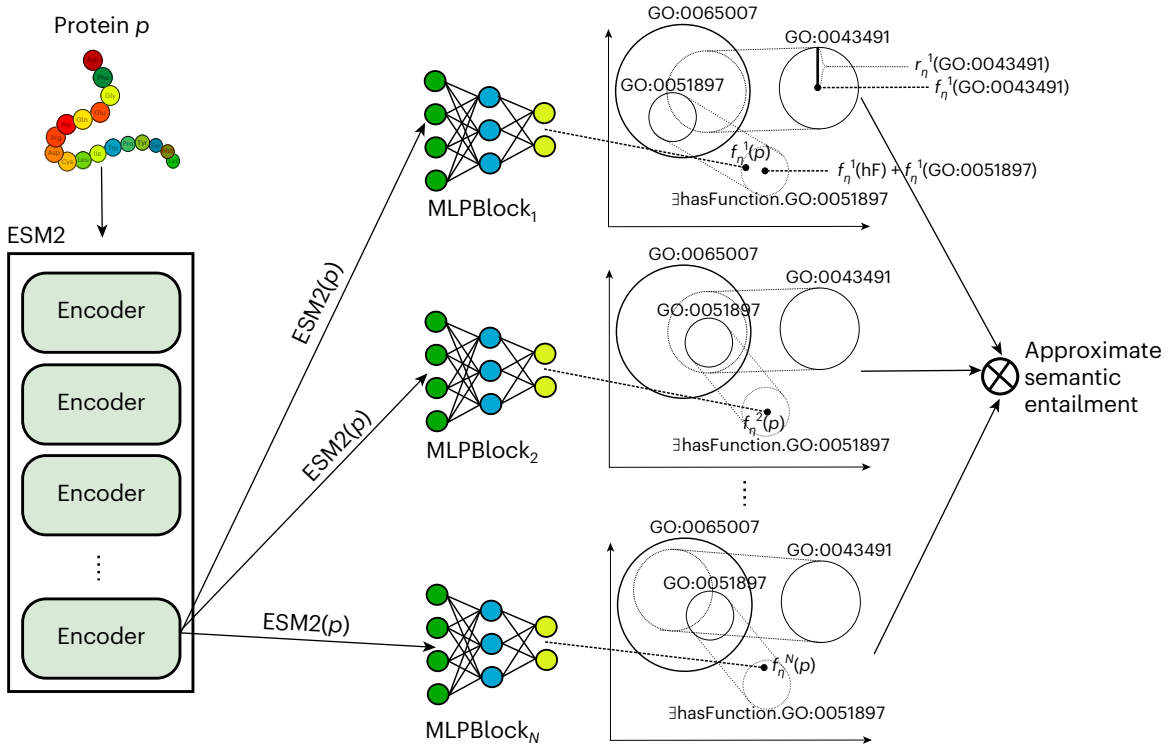

Fig. 1 | High-level overview of the DeepGO-SE model. Left: protein *p* is embedded in a vector space using ESM2 model. Right: multiple models with an MLP that embeds the protein in the same space as the GO axioms. Furthermore, predictions from multiple models are combined to perform approximate semantic entailment.

the precision–recall curve (AUPR), and class-centric area under the receiver operating characteristic curve (AUC) standardized by the Critical Assessment of Functional Annotation (CAFA) challenge[33,34]. We provide detailed information about evaluation measures in the Supplementary Information.

We train and evaluate three subontologies of GO separately because they have different characteristics in terms of the number of classes and their relations, the number of proteins and the sources of information they can benefit from. We compare with five baseline methods: Naïve, MLP, DeepGOCNN, DeepGOZero and DeepGraphGO. None of these methods relies on sequence similarity and, except for the naïve predictor, all assign functions based on sequence features that are learned directly or using features derived from tools such as InterProScan[35].

In all evaluations, the DeepGO-SE model significantly outperformed all the baseline methods in terms of *F* max, AUPR and AUC. In MFO, DeepGO-SE achieved an *F* max of 0.554, which is 7% larger than the result achieved by the MLP and DeepGOZero methods (Table 1). In predicting BPO annotations, the model achieves an *F* max of 0.432, which is around 8% higher than the best baseline method DeepGraphGO (Table 2), and in the CCO evaluation, DeepGO-SE model achieves an *F* max of 0.721 (Table 3).

In our basic DeepGO-SE model, protein embeddings are generated from protein sequence by ESM2; however, we can modify the protein embedding to encode more information about a protein. We argue that biological process and cellular component annotations cannot be predicted from a protein sequence alone because even sequence-identical proteins can legitimately be involved in different processes dependent on the presence or absence of other proteins. Therefore, we use the protein embedding to also encode information about a proteome and its interactions (protein–protein interactions, PPIs). We use this embedding function and alter the input vector to DeepGO-SE to perform three experiments. First, in DeepGOGAT-SE, we use the ESM2 embeddings as input for each protein. Second, in DeepGOGATMF-SE, the input consists of the experimental annotations of a protein to its molecular functions using a binary vector of size 6,851. Third, in DeepGOGATMF-SE-Pred, we use the prediction scores from the DeepGO-SE model for molecular functions as input. We train and evaluate these three models to determine the effect of incorporating interactions.

Table 1 | Prediction results for molecular functions on the UniProtKB/Swiss-Prot dataset

Method	*F* max	*S* min	AUPR	AUC
Naive	0.321	14.568	0.180	0.500
MLP	0.321	14.606	0.195	0.500
MLP (ESM2)	0.517	12.197	0.508	0.830
DeepGOCNN	0.404	13.741	0.365	0.749
DeepGOZero	0.483	12.722	0.444	0.749
DeepGraphGO	0.416	14.077	0.357	0.673
DeepGO-SE	**0.554**	11.681	**0.552**	**0.874**
DeepGOGAT-SE	0.525	**11.137**	0.523	0.861

This table shows protein-centric F_{max}, S_{min} and AUPR, and the class-centric average AUC. Bold values indicate best performance.

Combining PPIs and ESM2 embeddings in the DeepGOGAT-SE model reduced the MFO prediction performance to an *F* max of 0.525, but slightly improved *S* min. Incorporating PPIs improves the performance in BPO predictions to an *F* max of 0.435. The overall best performance in BPO is achieved when using experimental MFO annotations as features (DeepGOGATMF-SE), followed by MFO annotations predicted by DeepGO-SE (DeepGOGATMF-SE-Pred) (Table 2). For CCO, incorporating PPIs in the DeepGO-SE model increases *F* max from 0.721 to 0.736 (DeepGOGAT-SE) (Table 3).

Interestingly, including PPIs in our model did not improve MFO predictions (except for a slight improvement in *S* min), demonstrating

Table 2 | Prediction results for biological processes on the UniProtKB/Swiss-Prot dataset

Method	*F* max	*S* min	AUPR	AUC
Naive	0.294	43.934	0.195	0.500
MLP	0.295	43.914	0.210	0.499
MLP (ESM2)	0.423	39.721	0.388	0.864
DeepGOCNN	0.334	42.912	0.275	0.686
DeepGOZero	0.343	42.857	0.284	0.643
DeepGraphGO	0.354	42.100	0.303	0.736
DeepGO-SE	0.432	39.419	0.401	0.864
DeepGOGAT-SE	0.435	39.123	0.404	**0.876**
DeepGOGATMF-SE	**0.448**	**37.299**	**0.428**	0.831
DeepGOGATMF-SE-Pred	0.444	39.098	0.409	0.855

This table shows protein-centric F_{max}, S_{min} and AUPR, and the class-centric average AUC. Bold values indicate best performance.

Table 3 | Prediction results for cellular components on the UniProtKB/Swiss-Prot dataset

Method	*F* max	*S* min	AUPR	AUC
Naive	0.620	11.879	0.490	0.500
MLP	0.620	11.879	0.552	0.500
MLP (ESM2)	0.717	9.489	0.708	0.909
DeepGOCNN	0.661	11.079	0.670	0.758
DeepGOZero	0.625	11.700	0.587	0.599
DeepGraphGO	0.667	10.020	0.666	0.814
DeepGO-SE	0.721	9.499	0.730	0.914
DeepGOGAT-SE	**0.736**	**8.634**	0.743	**0.930**
DeepGOGATMF-SE	0.668	9.809	0.679	0.884
DeepGOGATMF-SE-Pred	0.694	9.907	**0.753**	0.884

This table shows protein-centric F_{max}, S_{min} and AUPR, and the class-centric average AUC. Bold values indicate best performance.

that molecular functions can be predicted from single proteins whereas information about multiple proteins needs to be used to predict BPO and CCO annotations.

neXtProt manual prediction dataset evaluation

In order to further evaluate the performance of our method and baseline methods, we used a dataset of manually predicted protein functions from neXtProt. neXtProt assigns functions to uncharacterized proteins based on expert curation of available evidence. We found that, for molecular functions, the best *F* max of 0.386 is achieved by our the DeepGO-SE method and the second best *F* max 0.382 is achieved by MLP (ESM2). Surprisingly, a similar performance is achieved by the Naïve method, which only uses the term frequency. However, when we evaluate based on AUPR and term centric AUC, we find that DeepGO-SE performs significantly better. The discrepancy can be explained by the small number of annotations. In this dataset, the median number of annotations is one, meaning that most proteins have only one specific GO function prediction (Table 4).

For biological processes, our method DeepGOGAT-SE which combines PPIs into the model performs with the best *F* max of 0.350. DeepGO-SE achieves a slightly lower *F* max of 0.349 and slightly better *S* min; however, DeepGOGAT-SE is substantially better in terms of AUPR and AUC. The third best *F* max and the best AUC are achieved by our method, which uses predicted molecular functions to predict biological processes. We were not able to evaluate the DeepGOGATMF-SE method because many of the proteins are missing manual molecular functions (Table 5). We also evaluated the statistical significance of the difference of the predictions for DeepGO-SE and DeepGOGAT-SE compared to the baseline methods (Supplementary Table D1) and find that DeepGO-SE performs significantly better than all baseline methods, and DeepGOGAT-SE performs better than all other methods in BPO and better than DeepGOZero, MLP and the Naïve predictor on MFO.

Validation based on structural homologues

We further investigated some predictions of molecular functions for which DeepGO-SE and neXtProt were in agreement to test if we could find additional evidence for the predictions. Specifically, we investigated the Mab-21-like protein 4 (MAB21L4) protein which has a single MFO annotation nucleotidyltransferase activity (GO:0016779), which is both assigned by neXtProt and DeepGO-SE. DeepGO-SE predicts this annotation with a high score of 0.638. MAB21L4 was predicted by neXtProt to be a nucleotidyltransferase based on available information about the protein's activity in epidermal keratinocytes[36]. As part of investigating the role of MAB21L4 in keratinocytes, distant homology detection was used to assign MAB21L4 to the nucleotidyltransferase (NTase) fold superfamily[37]. The active site is described by the motifs hG[GS], [DE]h[DE]h and h[DE]h (h indicates a hydrophobic amino acid), where the three conserved aspartate/glutamate are involved in coordination of divalent ions and activation of acceptor hydroxyl group of the substrate, and the hG[GS] pattern is involved in holding the substrates within the active site[37]. Sequence alignment combined with structural data provided by AlphaFold2 suggests that the [DE]h[DE]h motif is conserved in MAB21L4 (Asp80-Met81-Glu-82-Val83) while the h[DE]h motif that aligns with other family members may not be conserved as it is replaced by an histidine (Phe199-His200-Val201). An alternative h[DE]h motif is present in Val-236, Asp-237, Leu-238 with the two first residues in a loop and the third one at the beginning of a short β-strand. The hG[GS] motif is less conserved among the nucleotidyltransferase superfamily members and seems not conserved among the members of the Mab-21 group but is present in Mab-21-like protein 1 (MAB21L1); sequence-based methods like InterProScan identify a Mab-21-like, nucleotidyltransferase domain (IPR046903) in MAB21L1. We used foldseek[38] to compare MAB21L1 and MAB21L4 structurally, and found that both are structurally very similar despite a low sequence similarity. Furthermore, MAB21L4 is structurally very similar to Cyclic GMP-AMP synthase (CGAS), which is well-characterized as having the nucleotidyltransferase activity.

Another noticeable example is the Family With Sequence Similarity 151 Member B (FAM151B) protein which was predicted to be a phosphoric diester hydrolase (GO:0008081) based on structural similarity to a protein from *Sicarius terrosus* by the neXtProt database. DeepGO-SE predicted the same function with a high score of 0.846. Foldseek search resulted in many sequence and structure homologues. Structure homologues with high sequence identity were not annotated, however, we found several well annotated structural homologues with low sequence identity. For example, the human protein Lysophospholipase D (GDPD3) has a high structural similarity to FAM151B and has been annotated with phosphoric diester hydrolase activity (GO:0008081) based on experimental evidence (Supplementary Fig. C1). In addition, DeepGO-SE predicts other functions such as metal ion binding (GO:0046872) which GDPD3 has been annotated with as well. These findings suggest that DeepGO-SE learned to predict functions, among others, based on structural information.

Ablation study

In order to evaluate the contribution of the individual components of our models, we performed an ablation study. First, for each of the models, we removed the ElEmbeddings axiom loss functions and only optimized function prediction loss to determine the effect of using background knowledge contained in the GO. In the DeepGO-SE

Table 4 | Prediction results for molecular functions on the neXtProt dataset

Method	F max	S min	AUPR	AUC
Naive	0.360	10.340	0.165	0.500
MLP	0.347	10.371	0.194	0.493
MLP (ESM2)	0.382	**9.985**	0.292	0.730
DeepGOCNN	0.348	10.641	0.270	0.599
DeepGOZero	0.337	10.662	0.261	0.573
DeepGraphGO	0.330	10.573	0.270	0.558
TALE	0.344	10.673	0.238	0.640
SPROF-GO	0.352	10.331	0.270	0.652
DeepGO-SE	**0.386**	10.093	**0.324**	**0.744**
DeepGOGAT-SE	0.375	10.254	0.291	0.700

This table shows protein-centric F_{max}, S_{min} and AUPR, and the class-centric average AUC. Bold values indicate best performance.

Table 5 | Prediction results for biological processes on the neXtProt dataset

Method	F max	S min	AUPR	AUC
Naïve	0.308	32.987	0.183	0.500
MLP	0.310	32.033	0.206	0.502
MLP (ESM2)	0.336	**30.044**	0.305	0.682
DeepGOCNN	0.286	32.152	0.235	0.571
DeepGOZero	0.329	31.999	0.263	0.553
DeepGraphGO	0.322	31.861	0.240	0.558
TALE	0.280	32.973	0.221	0.533
SPROF-GO	0.312	31.164	0.251	0.620
DeepGO-SE	0.349	30.170	**0.312**	0.683
DeepGOGAT-SE	**0.350**	30.218	**0.312**	0.666
DeepGOGATMF-SE-Pred	0.339	30.653	0.293	**0.694**

This table shows protein-centric F_{max}, S_{min} and AUPR, and the class-centric average AUC. Bold values indicate best performance.

model, removing axioms losses resulted in a performance drop in the MFO evaluation while the performance in the BPO and CCO evaluations was not affected. Second, we trained the models with only GO or only GO-PLUS axioms to further evaluate the effect of using more background knowledge for performing approximate semantic entailment. We found that the performance of the MFO model improves with GO-PLUS axioms compared to GO axioms whereas the performance of the BPO and CCO models slightly drop when using the additional axioms contained in GO-PLUS.

Using PPI information, in the DeepGOGAT-SE model, removing axioms and removing the semantic entailment module resulted in a slight performance increase in MFO evaluation but the performance dropped in the BPO and CCO evaluations. In models that use PPIs and molecular functions as protein features, performance is better for BPO and CCO when removing axioms and semantic entailment.

Overall, the ablation study shows that the ontology axioms and semantic entailment mostly contribute to MFO and CCO model performance whereas the performance of BPO model is not significantly affected. The PPIs with GAT noticably contribute to CCO and BPO model performance and BPO model achieves the best performance without axioms and semantic entailment. Supplementary Table D2 provides the results of the ablation study for all four evaluation measures.

Discussion

DeepGO-SE is a protein function prediction method that improves the prediction performance for proteins by incorporating both protein sequence features generated by a pretrained protein language model, background knowledge from the GO and interactions between proteins. Our results allow us to draw three main conclusions: knowledge-enhanced machine learning methods are now able to improve over methods that do not rely on background knowledge; GO function prediction is best formulated using a separate, hierarchical prediction approach; and function prediction models based on ESM2 can now generalize to largely unseen proteins.

Although DeepGO-SE can predict biological processes and cellular components using only a protein sequence, the best performance is achieved when the sequence is combined with PPIs. However, many novel proteins do not have known interactions which limits the application of the combined model on them. Therefore, there is a need for methods which can accurately predict PPIs for novel proteins based on the only available sequence. In the future, we plan to incorporate sequence- and structure-based PPI predictors into our model.

In addition, DeepGO-SE is able to perform zero-shot predictions, similar to DeepGOZero, and is faster to obtain predictions than other methods that rely on multiple sequence alignments. This is due to the fact that DeepGO-SE relies only on ESM2 embeddings, which are faster to compute[30]. Overall, the DeepGO-SE model represents a significant improvement over existing protein function prediction methods, providing a more accurate, comprehensive and efficient approach.

Methods

UniProtKB/Swiss-Prot dataset

We use a dataset that was generated from manually curated and reviewed dataset of proteins from the UniProtKB/Swiss-Prot Knowledgebase[3] version 2021_04 released on 29 September 2021. We filtered all proteins with experimental functional annotations with evidence codes EXP, IDA, IPI, IMP, IGI, IEP, TAS, IC, HTP, HDA, HMP, HGI and HEP. The dataset contains 77,647 reviewed and manually annotated proteins. For this dataset we use Gene Ontology (GO) released on 16 November 2021. We train and evaluate models for each of the subontologies of GO separately.

We mainly aim to predict functions of novel proteins that have a low sequence similarity to existing proteins in the dataset. Therefore, we decided to split our dataset based on any similarity hit with a maximum e-value score of 0.001. We computed pairwise similarity using Diamond (v.2.0.9)[39] and grouped the sequences that have some similarity and split these groups into training, validation and testing sets. Supplementary Table D3 summarizes the datasets for each subontology. We train and evaluate a separate model for each subontology.

neXtProt dataset

In order to further evaluate the performance of our models we use a dataset of manually annotated predictions of uncharacterized human proteins from the neXtProt[40] database. neXtProt standardizes and integrates information on human proteins and provides users with an advanced search capability built around semantic technologies[40]. neXtProt contains free text summaries of the literature and standardized enzyme annotations from UniProtKB/Swiss-Prot, pathway annotations from KEGG[41] and Reactome[42], and GO MFO and BPO terms from a variety of resources, obtained either manually or by automatic procedures, and based on either experiments or computational analysis. Proteins lacking the above-mentioned annotations and those that are solely annotated with broad GO terms are considered as uncharacterized. They can be retrieved using the SPARQL[43] query NXQ_00022 (ref. 44). In the 18 April 2023 release of neXtProt, there are 1,521 such proteins. To stimulate the characterization of these poorly studied proteins, neXtProt collects and reviews functional predictions from the literature and proposes their own function annotations based on

a manual interpretation of different types of public data (phenotypes, expression, subcellular localization, protein and genetic interactions, phylogeny, structure, sequence and functional assays)[45]. These predictions are displayed in the function prediction pages as GO MFO or BPO terms, and the underlying evidence using the Evidence Code Ontology (ECO)[46].

Here we use the data retrieved from 113 publications together with different resources that were used to predict the functions of 239 uncharacterized human proteins. In total, the proteins collected 659 specific GO function annotations, where 69 molecular functions were assigned to 53 proteins and 590 biological processes were assigned to 225 proteins. Roughly one third of the proteins (38%) are assigned to only one function that in most of the cases (85%) is a GO BPO term. Most of the functional predictions (78%) are based on one piece of evidence.

Protein language model ESM2

Protein language models are large transformer architectures trained on protein sequences. The Evolutionary Scale Model (ESM)[30,47] has been trained on 250 million sequences and learned protein sequence representations that are predictive for biochemical and biological properties of proteins including their functions. The second version of ESM has been improved to learn better representations that are also predictive of tertiary structures of proteins. We use the pretrained model of ESM2 with 3 billion parameters (esm2_t36_3B_UR50D) to generate representations of proteins in our dataset. For a protein, we compute the output of the last layer and take the mean of embeddings for each amino acid, resulting in an embedding of size of 2,560 for each protein.

GO-PLUS

The standard version of GO does not include relations between GO classes and external ontologies such as ChEBI[48], Uberon[49], the Cell Ontology[50] or to structured vocabularies such as the NCBI Taxonomy[51]. These relations and cross-ontology axioms exist in an extended version called GO-PLUS[52]. For example, in GO-PLUS the class atrioventricular bundle cell differentiation (GO:0003167) is defined as equivalent to cell differentiation (GO:0030154) and results in acquisition of features of (RO:0002315) some atrioventricular bundle cell (CL:0010005). We use the GO-PLUS ontology version released on 16 Novemeber 2021 which has over 260K axioms. Like GO, GO-PLUS uses the Web Ontology Language (OWL) 2 (ref. 53) to represent its axioms. The Description Logic fragment of OWL 2, OWL 2 DL, defines several profiles, that is, restricted languages with specific computational properties. GO is formalized using the OWL EL profile[54]. However, GO-PLUS contains axioms that are not part of the OWL EL profile; therefore, it cannot directly be used with reasoning or machine learning methods that are based on OWL EL. We identify around 1,500 axioms that do not fit in the OWL EL profile and filtered them out using the EL Vira tool[55].

Approximate semantic entailment

Suppose $\mathcal{O}$ is an ontology composed of a set of class symbols $\mathbf{C}$, relation symbols $\mathbf{R}$ and individual symbols $\mathbf{I}$, and that it is expressed in the Description Logic $\mathcal{ALC}$ (ref. 56). In this logic, each class symbol is considered a class description. If C and D are class descriptions and R is a relation symbol, then the expressions $C \sqcap D$, $C \sqcup D$, $\neg C$, $\forall R.C$ and $\exists R.C$ are also considered as class descriptions.

In the $\mathcal{ALC}$ Description Logic, axioms can be classified as TBox or ABox axioms. If C and D are class descriptions, a and b are individual symbols, and r is a relation symbol, a TBox axiom has the form $C \sqsubseteq D$, while an ABox axiom has the form $C(a)$ or $r(a, b)$. A TBox is a set of TBox axioms, and an ABox is a set of ABox axioms. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ in $\mathcal{ALC}$ comprises a nonempty domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that satisfies $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ for all $C \in \mathbf{C}$, $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ for all $R \in \mathbf{R}$, and $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ for all $a \in \mathbf{I}$. The interpretation function is extended to concept descriptions as follows:

$$\begin{aligned}
&(C \sqcap D)^{\mathcal{I}} := C^{\mathcal{I}} \cap D^{\mathcal{I}}, (C \sqcup D)^{\mathcal{I}} := C^{\mathcal{I}} \cup D^{\mathcal{I}},\\
&(\forall R.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} | \forall e \in \Delta^{\mathcal{I}} : (d, e) \in R^{\mathcal{I}} \text{ implies } e \in C^{\mathcal{I}}\},\\
&(\exists R.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} | \exists e \in \Delta^{\mathcal{I}} : (d, e) \in R^{\mathcal{I}} \text{ and } e \in C^{\mathcal{I}}\},\\
&(\neg C)^{\mathcal{I}} := \Delta^{\mathcal{I}} - C^{\mathcal{I}}.
\end{aligned} \tag{1}$$

An interpretation $\mathcal{I}$ is called a model of a TBox if, for all $C \sqsubseteq D$ in the TBox, $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$; and a model of an ABox if, for all $R(a,b)$, $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$ and for all $C(a)$, $a^{\mathcal{I}} \in C^{\mathcal{I}}$.

A statement ϕ is semantically entailed by ontology $\mathcal{O}$ (consisting of TBox and ABox), denoted $\mathcal{O} \models \phi$, if and only if every model of $\mathcal{O}$ (that is, an interpretation $\mathcal{I}$ that is a model of both ABox and TBox of $\mathcal{O}$) is also a model of ϕ ($\mathrm{Mod}(\mathcal{O}) \subseteq \mathrm{Mod}(\phi)$). Semantic entailment requires access to all models of $\mathcal{O}$ which are usually infinite; approximate semantic entailment considers only a strict (usually finite) subset of $\mathrm{Mod}(\mathcal{O})$ and tests whether ϕ is true in each of them[26,57].

Here, we perform approximate semantic entailment by learning several models and determining whether a prediction (that is, a statement that assigns a function to a protein) is true in all of them. For each subontology of GO we train up to ten models and aggregate the prediction scores using three different strategies. First, we take the maximum of the selected scores which means that if the predictions is made, it is true in all generated models. Second, we take an average of the scores. Here, the prediction is made if the prediction threshold is lower than average of all models. Lastly, we take the minimum of the scores where we make sure that the prediction is true in at least one of the generated models. We select the best parameters of the approximate semantic entailment based on our validation set and use the same on our test set. Supplementary Tables D4–D7 summarize the results of semantic entailment on our validation set.

DeepGO-SE model

In the DeepGO-SE model, we use ESM2 (ref. 30) to represent a protein sequence and project them into multiple geometric interpretations (that is, models) of GO that have been generated with ELEmbeddings[31]; we then test the degree of truth of statements assigning a function to a protein in each interpretation of GO, and aggregate over all interpretations. The ESM2 embeddings of proteins are used as input to a multilayer perceptron (MLP) model that projects the embedding into the ELEmbeddings space by matching the dimensionality of the ESM2 embedding with the dimension of the ELEmbedding space:

$$f_\eta(p) = \mathrm{MLPBlock}(\mathrm{esm2}(p)) \tag{2}$$

Given a protein p and GO class c, we score the concept assertion statement $\exists \mathrm{hasFunction}.c(p)$ using the following formula:

$$y'_c = \mathrm{SE}_{i=1}^{N}(\sigma(f_\eta^i(p) \cdot (f_\eta^i(\mathrm{hF}) + f_\eta^i(c))^T + r_\eta^i(c))) \tag{3}$$

where $f_\eta^i(p)$ is the projection function from equation (2) in model i, $f_\eta^i(\mathrm{hF})$ is the embedding of the hasFunction relation in model i, $f_\eta^i(c)$ is the centre embedding of an n-ball representing class c in model i, $r_\eta^i(c))$ is the radius of the n-ball representing class c in model i, σ is a sigmoid activation function and $\mathrm{SE}_{i=1}^{N}$ is a function for performing approximate semantic entailment over N models.

To combine PPIs with individual features of proteins we use graph attention networks (GAT)[58] and embed the protein p in the ELEmbeddings space using the formula

$$f_\eta(p) = \mathrm{GATConv}(\mathrm{MLPBlock}(x), g) \tag{4}$$

where x is an input feature vector for p, g is the PPI graph, MLPBlock is described in equation (6), GATConv is a GAT layer.

The statement is approximately entailed if it is true in all interpretations generated by DeepGO-SE. We generate several ELEmbedding models and projection functions $f_\eta(p)$, and aggregate the truth values for the tested axiom in each of the models to obtain the final prediction scores (the degree of entailment). Given N interpretations, we aggregate the truth values using the function SE, which is either the minimum, maximum or arithmetic mean of the truth values in all N generated models. Figure 1 provides an overview of the prediction model of DeepGO-SE.

For each model, we compute the binary crossentropy loss between our predictions and the labels, and optimize them together with losses for ontology axioms from ELEmbeddings. We provide detailed descriptions of the ELEmbeddings loss functions in the Supplementary Information.

Protein–protein interaction networks

Molecular functions of proteins mainly depend on their sequences and structures. However, biological processes result from interactions between multiple proteins. Therefore, to accurately predict biological processes, it is necessary to include multiple proteins and their interactions.

For our experiments, we use functional interactions between proteins provided by the STRING database (v.11.0)[59]. We filter out all the interactions with confidence score less than 0.7. Our dataset uses UniProtKB identifiers and we map them to STRING database identifiers with mappings provided by UniProtKB. We generate the protein interaction graph using all the proteins in our dataset and use the DGL[60] library to process it and train graph neural networks.

Baseline methods

For our evaluations we selected methods that do not rely on predictions based on sequence similarity because our aim is to test the predictors on novel sequences. Therefore, we do not include methods as baselines that are primarily based on sequence similarity, such as predictions using BLAST or Diamond, or any other predictors that use their combinations.

Naive approach. Due to the imbalance in GO class annotations and propagation based on the true-path-rule, some classes have more annotations than others. Therefore, it is possible to obtain prediction results just by assigning the same GO classes to all proteins based on annotation frequencies. In order to test the performance obtained based on annotation frequencies, CAFA introduced a baseline approach called 'naive' classifier[34]. Here, each query protein p is annotated with the GO classes with a prediction score computed as:

$$S(p,f) = \frac{N_f}{N_{\text{total}}} \tag{5}$$

where f is a GO class, N_f is a number of training proteins annotated by GO class f and N_{total} is a total number of training proteins. We implement the same method.

MLP. The MLP and MLP (ESM2) baseline methods predict protein functions using a multilayer perceptron (MLP) from a protein's InterPro domain annotations obtained with InterProScan[35] and ESM2 (ref. 30) embeddings. We represent a protein with a binary vector for all the InterPro domains or ESM2 embeddings and pass it to two layers of MLP blocks where the output of the second MLP block has residual connection to the first block. This representation is passed to the final classification layer with sigmoid activation function. One MLP block performs the following operations:

$$\text{MLPBlock}(\mathbf{x}) = \text{DropOut}(\text{BatchNorm}(\text{ReLU}(W\mathbf{x} + b))) \tag{6}$$

The input vector $\mathbf{x}$ of length 26,406 represents InterPro domain annotations or ESM2 embedding. It is reduced to 1,024 by the first MLPBlock:

$$\mathbf{h} = \text{MLPBlock}(\mathbf{x}) \tag{7}$$

This representation is passed to the second MLPBlock with the input and output size of 1,024 and added to itself using residual connection:

$$\mathbf{h} = \mathbf{h} + \text{MLPBlock}(\mathbf{h}) \tag{8}$$

Finally, we pass this vector to a classification layer with a sigmoid activation function. The output size of this layer is the same as the number of classes in each subontology:

$$\mathbf{y} = \sigma(W\mathbf{h} + b) \tag{9}$$

We train a different model for each subontology in GO.

DeepGOPLUS and DeepGOCNN. DeepGOPLUS[13] predicts function annotations of proteins by combining DeepGOCNN, which predicts functions from the amino acid sequence of a protein using a one-dimensional convolutional neural network (CNN), with the DiamondScore method. DeepGOCNN captures sequence motifs that are related to GO functions. Here, we only use CNN based predictions.

DeepGOZero. DeepGOZero[11] combines protein function prediction with a model-theoretic approach for embedding ontologies into a distributed space, ELEmbeddings[31]. ELEmbeddings represent classes as n-balls and relations as vectors to embed ontology semantics into a geometric model. It uses InterPro domain annotations represented as binary vector as input and applies two layers of MLPBlock as in our MLP baseline method to generate an embedding of size 1,024 for a protein. It learns the embedding space for GO classes using ELEmbeddings loss functions and optimizes together with protein function prediction loss. For a given protein p, DeepGOZero predicts annotations for a class c using the following formula:

$$y'_c = \sigma(f_\eta(p) \cdot (f_\eta(\text{hF}) + f_\eta(c))^T + r_\eta(c)) \tag{10}$$

where f_η is an embedding function, hF is the hasFunction relation, $r_\eta(c)$ is the radius of an n-ball for a class c and σ is a sigmoid activation function. It optimizes binary crossentropy loss between predictions and the labels together with ontology axioms losses from ELEmbeddings.

DeepGraphGO. The DeepGraphGO[6] method uses a neural network to combine sequence features (InterPRO domain annotations) with PPI networks by using graph convolutional neural networks. We have implemented DeepGraphGO based on the manuscript and provide the source code for our implementation. We trained and evaluated the model using our UniProtKB/Swiss-Prot dataset.

TALE. TALE[14] predicts functions using a transformer-based deep neural network model which incorporates hierarchical relations from the GO into the model's loss function. The deep neural network predictions are combined with predictions based on sequence similarity. We used the trained models provided by the authors to evaluate them on the neXtProt dataset.

SPROF-GO. SPROF-GO[29] method uses the ProtT5-XL-U50 (ref. 61) protein language model to extract proteins sequence embeddings and learns an attention-based neural network model. The model incorporates the hierarchical structure of GO into the neural network and predicts functions that are consistent with hierarchical relations of GO classes. Furthermore, SPROF-GO combines sequence similarity-based predictions using a homology-based label diffusion

algorithm. We used the trained models provided by the authors to evaluate them on the neXtProt dataset.

Data availability

All data underlying this work is freely available at https://doi.org/10.5281/zenodo.10369249 (ref. 62).

Code availability

The source code of this work is freely available at https://github.com/bio-ontology-research-group/deepgo2 (ref. 63).

References

1. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
2. Ashburner, M. et al. Gene ontology: tool for the unification of biology. *Nat. Genet.* **25**, 25–29 (2000).
3. Consortium, T. U. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Res.* **51**, D523–D531 (2022).
4. Zhou, N. et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.* **20**, 244 (2019).
5. You, R. et al. GOLabeler: improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics* **34**, 2465–2473 (2018).
6. You, R., Yao, S., Mamitsuka, H. & Zhu, S. DeepGraphGO: graph neural network for large-scale, multispecies protein function prediction. *Bioinformatics* **37**, i262–i271 (2021).
7. You, R. et al. NetGO: improving large-scale protein function prediction with massive network information. *Nucleic Acids Res.* **47**, W379–W387 (2019).
8. You, R., Huang, X. & Zhu, S. Deeptext2go: improving large-scale protein function prediction with deep semantic text representation. *Methods* **145**, 82–90 (2018).
9. Gligorijević, V. et al. Structure-based protein function prediction using graph convolutional networks. *Nat. Commun.* **12**, 3168 (2021).
10. Lai, B. & Xu, J. Accurate protein function prediction via graph attention networks with predicted structure information. *Brief. Bioinform.* **23**, Bbab502 (2021).
11. Kulmanov, M. & Hoehndorf, R. DeepGOZero: improving protein function prediction from sequence and zero-shot learning based on ontology axioms. *Bioinformatics* **38**, i238–i245 (2022).
12. Kulmanov, M., Khan, M. A. & Hoehndorf, R. DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics* **34**, 660–668 (2017).
13. Kulmanov, M. & Hoehndorf, R. DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics* **36**, 442–449 (2019).
14. Cao, Y. & Shen, Y. TALE: Transformer-based protein function Annotation with joint sequence-Label Embedding. *Bioinformatics* **37**, 2825–2833 (2021).
15. Pan, T. et al. PFresGO: an attention mechanism-based deep-learning approach for protein annotation by integrating gene ontology inter-relationships. *Bioinformatics* **39**, Btad094 (2023).
16. Wu, Z., Guo, M., Jin, X., Chen, J. & Liu, B. CFAGO: cross-fusion of network and attributes based on attention mechanism for protein function prediction. *Bioinformatics* **39**, Btad123 (2023).
17. Wekesa, J. S., Luan, Y. & Meng, J. Predicting protein functions based on differential co-expression and neighborhood analysis. *J. Comput. Biol.* **28**, 1–18 (2021).
18. Makrodimitris, S., Reinders, M. J. T. & van Ham, R. C. H. J. Metric learning on expression data for gene function prediction. *Bioinformatics* **36**, 1182–1190 (2020).
19. Pellegrini, M. Using phylogenetic profiles to predict functional relationships. *Methods Mol. Biol.* **804**, 167–177 (2012).
20. Nevers, Y. et al. Insights into ciliary genes and evolution from multi-level phylogenetic profiling. *Mol. Biol. Evol.* **34**, 2016–2034 (2017).
21. Yao, S. et al. NetGO 2.0: improving large-scale protein function prediction with massive sequence, text, domain, family and network information. *Nucleic Acids Res.* **49**, W469–W475 (2021).
22. Krissinel, E. On the relationship between sequence and structure similarities in proteomics. *Bioinformatics* **23**, 717–723 (2007).
23. Hoehndorf, R., Schofield, P. N. & Gkoutos, G. V. The role of ontologies in biological and biomedical research: a functional perspective. *Brief. Bioinform.* **16**, 1069–1080 (2015).
24. Chowdhury, T. et al. Knowledge-enhanced neural machine reasoning: a review. Preprint at https://arxiv.org/abs/2302.02093 (2023).
25. Kulmanov, M., Smaili, F. Z., Gao, X. & Hoehndorf, R. Semantic similarity and machine learning with ontologies. *Brief. Bioinform.* **22**, bbaa199 (2020).
26. Tang, Z., Hinnerichs, T., Peng, X., Zhang, X. & Hoehndorf, R. FALCON: faithful neural semantic entailment over ALC ontologies. Preprint at https://arxiv.org/abs/2208.07628 (2023).
27. Kahanda, I. & Ben-Hur, A. Gostruct 2.0: automated protein function prediction for annotated proteins. In *Proc. of the 8th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics* (eds Haspel, N. et al.) 60–66 (Association for Computing Machinery, 2017).
28. Sureyya Rifaioglu, A., Doğan, T., Jesus Martin, M., Cetin-Atalay, R. & Atalay, V. Deepred: automated protein function prediction with multi-task feed-forward deep neural networks. *Sci. Rep.* **9**, 7344 (2019).
29. 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.* **24**, bbad117 (2023).
30. Lin, Z. et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Sci. Rep.* **379**, 1123–1130 (2023).
31. Kulmanov, M., Liu-Wei, W., Yan, Y. & Hoehndorf, R. El embeddings: geometric construction of models for the description logic el++. In *Proc. of the 28th International Joint Conference on Artificial Intelligence* (ed. Kraus, S.) 6103–6109 (International Joint Conferences on Artificial Intelligence Organization, 2019).
32. Henkin, L., Suppes, P. & Tarski, A. The axiomatic method with special reference to geometry and physics. In *Proc. of the International Symposium on the Axiomatic Method* 1–488 (North-Holland, 1959).
33. Radivojac, P. & Clark, W. T. Information-theoretic evaluation of predicted ontological annotations. *Bioinformatics* **29**, i53–i61 (2013).
34. Radivojac, P. et al. A large-scale evaluation of computational protein function prediction. *Nat. Meth.* **10**, 221–227 (2013).
35. Mitchell, A. et al. InterProScan 5: genome-scale protein function classification. *Bioinformatics* **30**, 1236–1240 (2014).
36. Ogami, T. et al. Mab21l4 regulates the tgf-β-induced expression of target genes in epidermal keratinocytes. *J. Biochem.* **171**, 399–410 (2022).
37. Kuchta, K., Knizewski, L., Wyrwicz, L. S., Rychlewski, L. & Ginalski, K. Comprehensive classification of nucleotidyltransferase fold proteins: identification of novel families and their representatives in human. *Nucleic Acids Res.* **37**, 7701–7714 (2009).
38. van Kempen, M. et al. Fast and accurate protein structure search with foldseek. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-023-01773-0 (2023).
39. Buchfink, B., Xie, C. & Huson, D. H. Fast and sensitive protein alignment using diamond. *Nat. Methods* **12**, 59–60 (2015).
40. Zahn-Zabal, M. et al. The neXtProt knowledgebase in 2020: data, tools and usability improvements. *Nucleic Acids Res.* **48**, D328–D334 (2019).

41. Kanehisa, M., Furumichi, M., Tanabe, M., Sato, Y. & Morishima, K. KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Res.* **45**, D353–D361 (2016).
42. Gillespie, M. et al. The reactome pathway knowledgebase 2022. *Nucleic Acids Res.* **50**, D687–D692 (2021).
43. Seaborne, A. & Prud'hommeaux, E. SPARQL query language for RDF. *W3C* www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/ (2008).
44. Duek, P., Gateau, A., Bairoch, A. & Lane, L. Exploring the uncharacterized human proteome using nextprot. *J. Proteome Res.* **17**, 4211–4226 (2018).
45. Duek, P., Mary, C., Zahn-Zabal, M., Bairoch, A. & Lane, L. Functionathon: a manual data mining workflow to generate functional hypotheses for uncharacterized human proteins and its application by undergraduate students. *Database* 2021, Baab046 (2021).
46. Nadendla, S. et al. ECO: the Evidence and Conclusion Ontology, an update for 2022. *Nucleic Acids Res.* **50**, D1515–D1521 (2021).
47. Rives, A. et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl. Acad. Sci. USA* **118**, e2016239118 (2021).
48. Degtyarenko, K. et al. ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Res.* **36**, D344–D350 (2007).
49. Mungall, C., Torniai, C., Gkoutos, G., Lewis, S. & Haendel, M. Uberon, an integrative multi-species anatomy ontology. *Genome Biol.* **13**, R5 (2012).
50. Diehl, A. D. et al. The cell ontology 2016: enhanced content, modularization, and ontology interoperability. *J. Biomed. Semant.* **7**, 44 (2016).
51. Schoch, C. L. et al. NCBI Taxonomy: a comprehensive update on curation, resources and tools. *Database* **2020**, Baaa062 (2020).
52. Consortium, T. G. O. The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Res.* **49**, D325–D334 (2020).
53. Grau, B. et al. OWL 2: the next step for OWL. *J. Web. Semant.* **6**, 309–322 (2008).
54. Motik, B. et al. OWL 2 web ontology language profiles. *W3C* https://www.w3.org/TR/owl2-profiles/ (2012).
55. Hoehndorf, R. et al. A common layer of interoperability for biomedical ontologies based on OWL EL. *Bioinformatics* **27**, 1001–1008 (2011).
56. Baader, F., Calvanese, D., McGuinness, D., Nardi, D. & Patel-Schneider, P. *The Description Logic Handbook: Theory, Implementation and Applications* (Cambridge Univ. Press, 2003).
57. Cadoli, M. & Schaerf, M. in *Trends in Artificial Intelligence* (eds Ardizzone, E., Gaglio, S. & Sorbello, F.) 68–77 (Springer, 1991).
58. Veličković, P. et al. Graph attention networks. In *6th International Conference on Learning Representations (ICLR)* (2018); https://openreview.net/forum?id=rJXMpikCZ
59. Szklarczyk, D. et al. STRING v11: protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res.* **47**, D607–D613 (2018).
60. Wang, M. et al. Deep graph library: a graph-centric, highly-performant package for graph neural networks. Preprint at https://arxiv.org/abs/1909.01315 (2019).
61. Elnaggar, A. et al. Prottrans: toward understanding the language of life through self-supervised learning. *IEEE Trans. on Pattern Anal. and Mach. Intell.* **44**, 7112–7127 (2022).
62. Kulmanov, M. Deepgo-se protein function prediction model data. *Zenodo* https://doi.org/10.5281/zenodo.10369249 (2023).
63. Kulmanov, M. & Zhapa, F. bio-ontology-research-group/deepgo2: v1.0.0. *Zenodo* https://doi.org/10.5281/zenodo.10369694 (2023).

Acknowledgements

This work has been supported by funding from the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award Nos. REI/1/5235-01-01, URF/1/4675-01-01, URF/1/4697-01-01, URF/1/5041-01-01 and FCC/1/1976-46-01. This work was supported by the SDAIA-KAUST Center of Excellence in Data Science and Artificial Intelligence (SDAIA-KAUST AI). For computer time, this research used the resources of the Supercomputing Laboratory at the King Abdullah University of Science & Technology (KAUST) in Thuwal, Saudi Arabia.

Author contributions

R.H. and M.K. conceived and designed the research. R.H. led the study and M.K. developed the prediction model and evaluated it. P.D.R. and L.L. contributed to functional analysis. F.J.G.V. and S.T.A. contributed to the structural analysis. R.H. and M.K. developed the first draft of the paper. All authors contributed to writing and improving the paper and approved the submission.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-024-00795-w.

Correspondence and requests for materials should be addressed to Maxat Kulmanov or Robert Hoehndorf.

Peer review information *Nature Machine Intelligence* thanks Marco Falda and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.