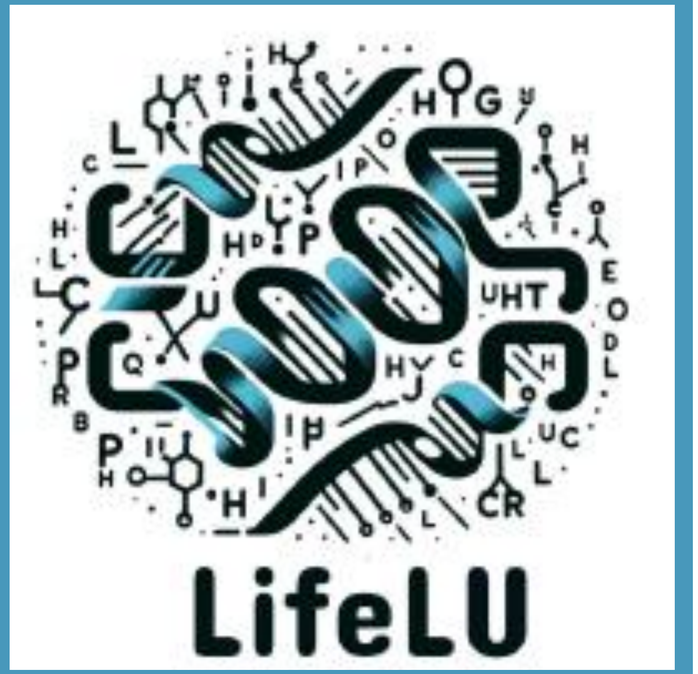
LifeLU - UNDERSTANDING THE LANGUAGE OF LIFE

READING GROUP



Presenter: Özlem Şimşek

Paper: TransFun: Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function

Link: https://academic.oup.com/bioinformatics/article/39/Supplement_1/i318/7210446?login=false

Date: 07/11/2024

Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function

Boadu, F., Cao, H., & Cheng, J. (2023). Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function. Bioinformatics, 39, i318–i325.

Bioinformatics, 2023, **39**, i318–i325
https://doi.org/10.1093/bioinformatics/btad208
ISMB/ECCB 2023

OXFORD

Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function

Frimpong Boadu[1], Hongyuan Cao[2], Jianlin Cheng[1,*]

[1]Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211, United States
[2]Department of Statistics, Florida State University, Tallahassee, FL 32306, Unites States

*Corresponding author. Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211, USA.
E-mail: chengji@missouri.edu

Abstract

Motivation: Millions of protein sequences have been generated by numerous genome and transcriptome sequencing projects. However, experimentally determining the function of the proteins is still a time consuming, low-throughput, and expensive process, leading to a large protein sequence-function gap. Therefore, it is important to develop computational methods to accurately predict protein function to fill the gap. Even though many methods have been developed to use protein sequences as input to predict function, much fewer methods leverage protein structures in protein function prediction because there was lack of accurate protein structures for most proteins until recently.

Results: We developed TransFun—a method using a transformer-based protein language model and 3D-equivariant graph neural networks to distill information from both protein sequences and structures to predict protein function. It extracts feature embeddings from protein sequences using a pre-trained protein language model (ESM) via transfer learning and combines them with 3D structures of proteins predicted by AlphaFold2 through equivariant graph neural networks. Benchmarked on the CAFA3 test dataset and a new test dataset, TransFun outperforms several state-of-the-art methods, indicating that the language model and 3D-equivariant graph neural networks are effective methods to leverage protein sequences and structures to improve protein function prediction. Combining TransFun predictions and sequence similarity-based predictions can further increase prediction accuracy.

Availability and implementation: The source code of TransFun is available at https://github.com/jianlin-cheng/TransFun.

1 Introduction

Proteins are essential macromolecules that carry out critical functions such as catalyzing chemical reactions, regulating gene expression, and passing molecular signals in living systems. It is critical to elucidate the function of proteins. However, even though various next-generation genome and transcriptome sequencing projects have generated millions of protein sequences, the experimental determination of protein function is still a low-throughput, expensive and time-consuming process. Thus, there is a huge gap between the number of proteins with known sequence and the number of proteins with known function, and this gap keeps increasing. As a result, it is important to develop computational methods to accurately predict the function of proteins.

Given the sequence of a protein and/or other information as input, protein function prediction methods aim to assign the protein to one or more function terms defined by Gene Ontology (GO) (Huntley et al. 2015). GO organizes function terms into three ontology categories: Biological Process (BP), Molecular Function (MF) and Cellular Component (CC). The terms in each of these ontology categories can be represented as a directed acyclic graph, in which parent nodes denoting broader (more general) function terms point to child nodes denoting more specific function terms.

Many protein function prediction methods use sequence or structure similarity to predict function, assuming proteins with similar sequences and structures likely have similar function. For example, GOtcha, Blast2GO (Martin et al. 2004; Conesa and Götz 2008), PDCN (Wang et al. 2013), and DIAMONDScore use sequence alignment methods such as BLAST (Altschul et al. 1997) to search for homologous sequences with known function for a target protein and then transfer their known function to the target. COFACTOR and ProFunc (Laskowski et al. 2005; Zhang et al. 2017) use structure alignment to search for function-annotated proteins whose structures are similar to the target protein to transfer the function annotation. There are also some methods leveraging interactions between proteins or co-expression between genes to predict function, assuming that the proteins that interact or whose genes have similar expression patterns may have similar function. For instance, NetGO (You et al. 2019) transfers to a target protein the known function of the proteins that interact with it. All these nearest neighbor-based methods depend on finding related function-annotated proteins (or called templates) according to sequence similarity, structure similarity, gene expression similarity, or protein–protein interaction, which are often not available. Therefore, they cannot generally achieve high-accuracy protein function prediction for most proteins.

To improve the generalization capability of protein function prediction, advanced machine learning-based methods such as FFPred and labeler (Cozzetto et al. 2016; You et al. 2018) have been developed to directly predict the function of a protein from its sequence. However, most of these methods

SUMMARY

Aim: predict GO terms from protein sequences and AlphaFold models (uses tertiary structures)

Problem type: multi-label classification

Method: combination of ESM language model and equivariant graph neural networks

GLOSSARY

ESM language model

???

GLOSSARY

Equivariant Graph Neural Networks (EGNN)

- Capture the essential features of protein structures that are invariant to the rotation and translation of 3D protein structures to improve protein function.

SUMMARY

Aim: predict GO terms from protein sequences and AlphaFold models (uses tertiary structures)

Problem type: multi-label classification

Method: combination of ESM language model and equivariant graph neural networks

DATASET

- UniProt\Swiss-Prot
  - protein sequences with function annotations - 566 996
- Open Biological and Biomedical Ontology (OBO)
  - the ontology graph
- AlphaFold DB
  - the predicted protein tertiary structures - 542 380
- All three ontologies of GO terms
  - molecular function ontology (MFO)
  - biological process ontology (BPO)
  - cellular component ontology (CCO)
- Proteins with sequence length between 100 and 1022
- Removed all 3328 CAFA3 test proteins and any protein that has 50% sequence identity with any protein in the CAFA3 test dataset
- GO terms that have at least 60 proteins for training and test

- CAFA3 test dataset
- Second test dataset: new proteins released between March 2022 and November 2022 in UniProt

DATASET

Table 1. The statistics of the curated protein function prediction dataset.[a]

Ontology	No. of protein	No. GO terms	Sequence identity threshold			
			0.3	0.5	0.9	0.95
MF	35 507	600	14 667	19 512	26 876	28 067
CC	50 340	547	20 679	26 808	36 721	38 509
BP	50 320	3774	20 180	26 647	37 536	39 348

[a] The first three columns are the GO ontology category, the total number of proteins in each category and the number of GO terms in each category. The remaining four columns list the number of protein clusters at each sequence identity threshold (0.3, 0.5, 0.9, 0.95).

METHOD

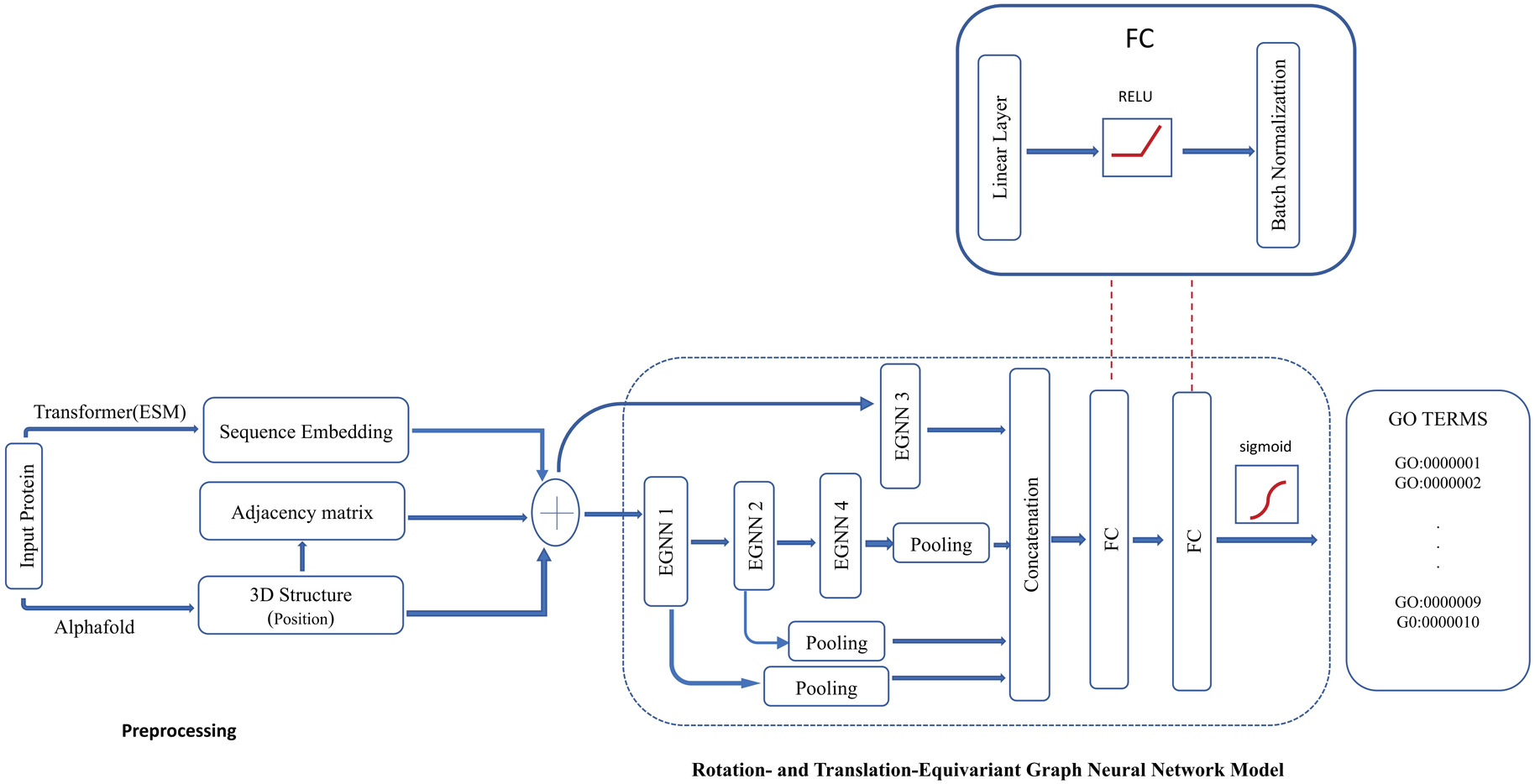


Figure 1. The protein function prediction pipeline of TransFun. The pipeline is divided into two main components, feature preprocessing (left) and neural network model (right). The input is a protein sequence. The output is the predicted probabilities of the GO terms for the protein.

METHOD

Features:

- ✦ Protein graph from a predicted structure (PDB)
- ✦ Embeddings from a protein sequence

METHOD

Protein graph extraction from predicted structure:

- The nodes in the graph represent residues of the protein.
- 2 different edge strategies:
  - a distance threshold approach
    - the Euclidean distance between carbon alpha atoms of two neighboring residues
    - tested 4,6,8,10 and 12A˚and chose 10A˚ as final distance threshold
  - a KNN approach
    - tested $\sqrt{n}$ and $\sqrt[3]{n}$ and chose $\sqrt[3]{n}$ as K where n is the number of residues
  - No self-loops

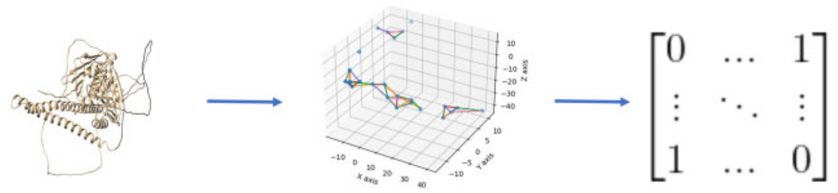


Figure 2. Constructing a graph from a protein structure. A graph is constructed by a K-nearest neighbor approach. The graph is stored in a binary adjacency matrix.

METHOD

Sequence feature extraction using transformer language model:

- ESM-1b
  - Per-residue embeddings (dimension: n x 1022)
  - Per-sequence embeddings (dimension: 1022)
    - an aggregation over the per-residue embeddings
  - from the 33rd layer

METHOD

Model:

- Rotation- and translation-equivariant graph neural network (EGNN):
  - 4 blocks of EGNNs where each EGNN has 4 layers
  - EGNN1:
    - Input: the protein graph with the per-residue embeddings
    - Output: a new per-residue embedding of dimension C

  (C: the number of GO classes to be predicted)
  - EGNN2:
    - Input: the protein graph output of EGNN1
    - Output: a new per-residue embedding of dimension C/2

METHOD

Model:

- Rotation- and translation-equivariant graph neural network (EGNN):
  - EGNN3:
    - Input: the protein graph with the per-sequence embeddings
    - Output: a new per-sequence embedding of dimension C/2

  (C: the number of GO classes to be predicted)
  - EGNN4:
    - Input: the protein graph output of EGNN2
    - Output: a new embedding of dimension C/4

METHOD

Model:

- Rotation- and translation-equivariant graph neural network (EGNN):
  - Aggregation:
    - The output embeddings from EGNN1, EGNN2, and EGNN4 are aggregated using a global mean pooling
    - The result is then concatenated with EGNN3 output
    - The concatenated features are then passed through 2 FC layers and RELU
    - Final node feature dimension is C
    - Ablation study in Supplementary Section

METHOD

Class Imbalance problem:

- The numbers of examples for different GO terms are very different.
- Use class weights to scale the training loss for GO terms appropriately
- Weigh less-represented GO terms more

RESULTS

Table 2. The results of TransFun and several other methods on the CAFA3 test dataset.[a]

Method	F_{max}			AUPR		
	MF	CC	BP	MF	CC	BP
Naive	0.295	0.539	0.315	0.138	0.373	0.197
DIAMONDScore	0.532	0.523	0.382	0.461	0.5	0.304
DeepGO	0.392	0.502	0.362	0.312	0.446	0.213
DeepGOCNN	0.411	0.582	0.388	0.402	0.523	0.213
TALE	0.548	0.654	**0.398**	0.485	**0.649**	0.258
TransFun	**0.551**	**0.659**	0.395	**0.489**	0.634	**0.333**

[a] TransFun was pretrained on the curated dataset whose proteins were clustered at sequence identity threshold of 50%. Bold numbers denote the best results.

- Performance on CAFA3 test dataset

RESULTS

Table 3. The results of TransFun on the test datasets having different identity thresholds with respect to the training data.

Score	30%			50%			90%		
	MF	CC	BP	MF	CC	BP	MF	CC	BP
Fmax	0.509	0.619	0.394	0.53	0.631	0.37	0.53	0.606	0.367
AUPR	0.461	0.599	0.333	0.489	0.614	0.327	0.487	0.61	0.3

- Impact of sequence identity on CAFA3 test dataset

RESULTS

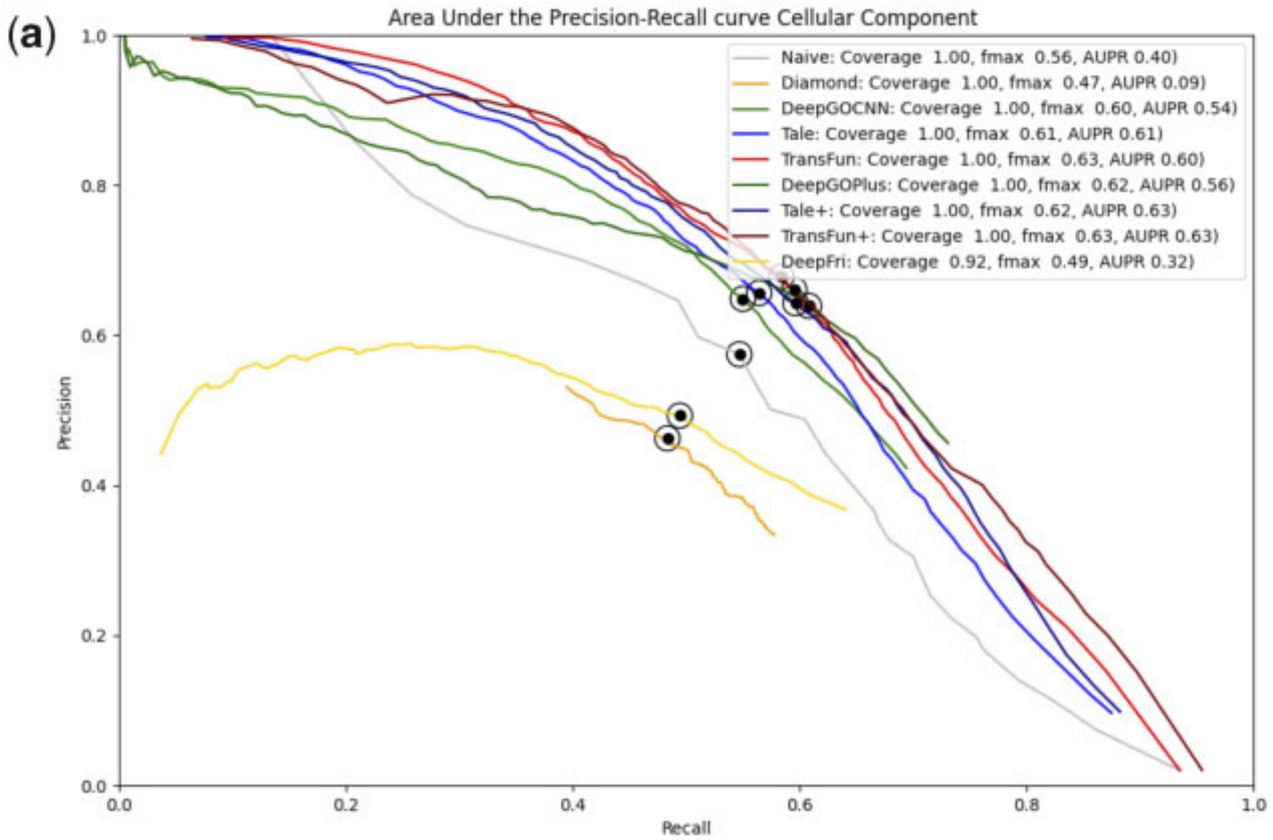
Table 4. The results on the new test dataset.[a]

Method	F_{max}			AUPR		
	CC	MF	BP	CC	MF	BP
Naïve	0.560	0.275	0.283	0.404	0.135	0.173
Diamond	0.473	0.564	0.392	0.089	0.115	0.080
DeepGOCNN	0.595	0.440	0.361	0.545	0.307	0.240
TALE	0.607	0.512	0.344	**0.613**	0.480	0.257
DeepFRI	0.494	0.454	0.324	0.324	0.303	0.169
TransFun	**0.628**	**0.608**	**0.413**	0.603	**0.569**	**0.366**
DeepGOPlus	0.623	0.635	**0.460**	0.562	0.549	0.339
TALE+	0.619	0.635	0.431	**0.633**	0.613	0.344
TransFun+	**0.628**	**0.638**	0.452	0.627	**0.638**	**0.410**

[a] Naïve, Diamond, DeepGOCNN, TALE, DeepFRI and TransFun (green) are individual methods. DeepGOPlus, TALE+ and TransFun+ (blue) are composite methods. The best results of among the individual methods or among the composite methods are bold.

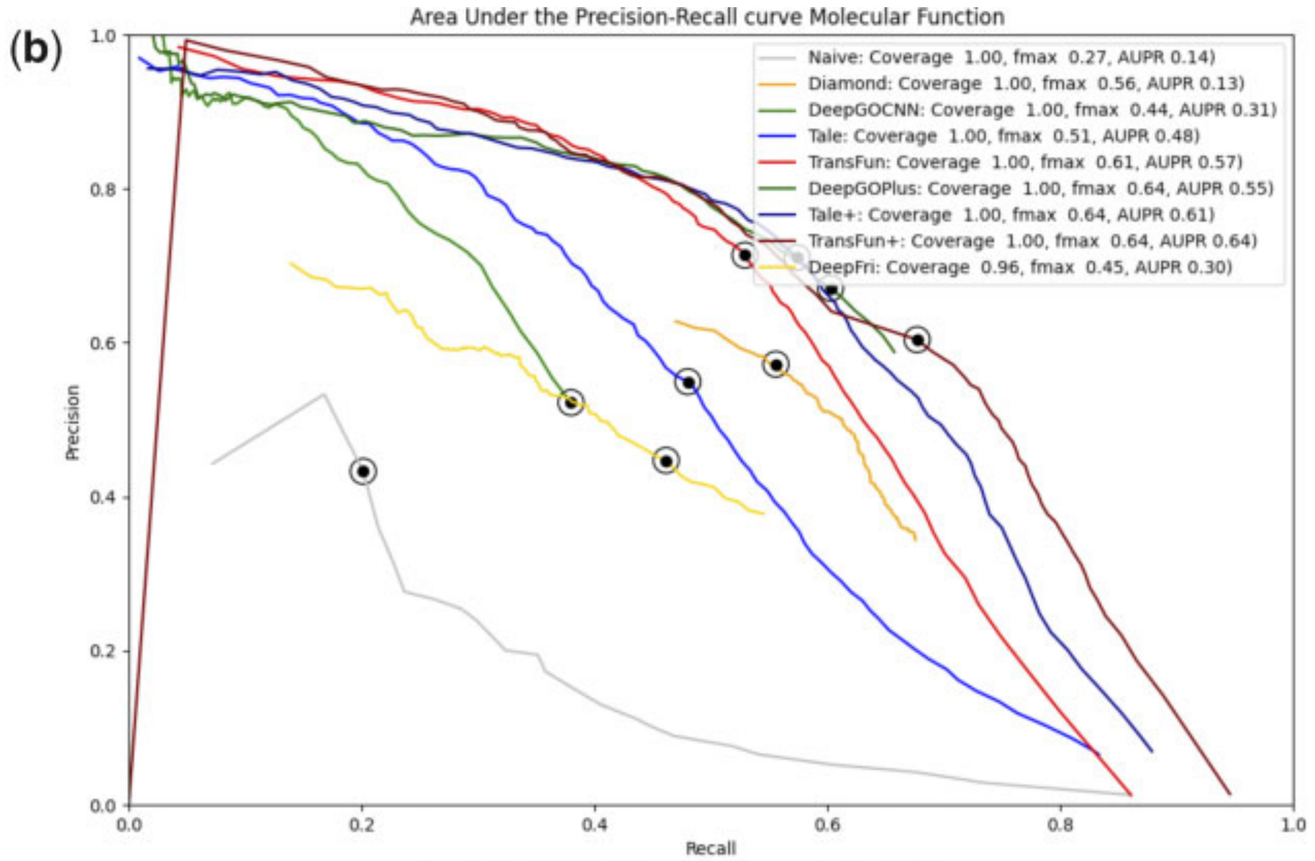
- Performance on the new test dataset

RESULTS



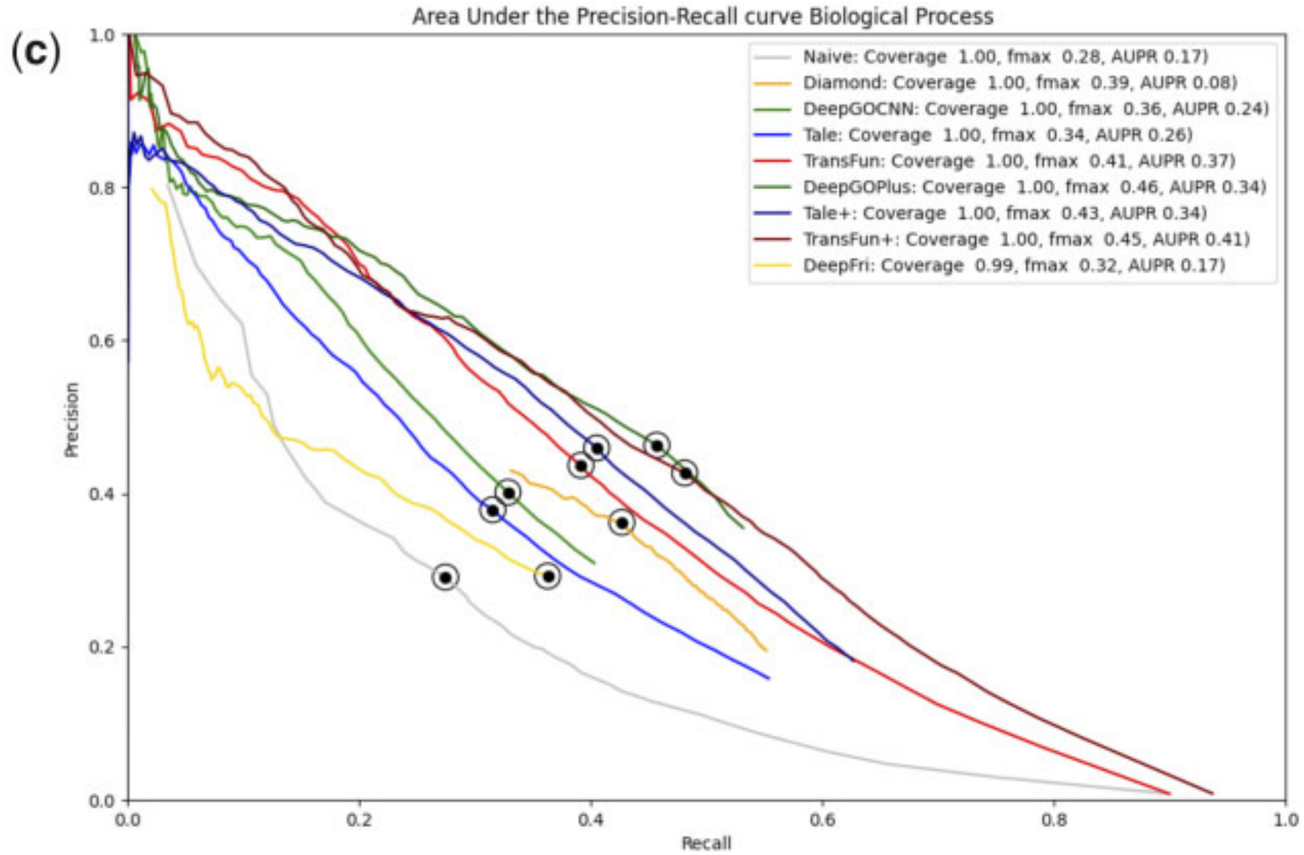
- Precision-recall curves on the new test dataset (CC)

RESULTS



- Precision-recall curves on the new test dataset (MF)

RESULTS



- Precision-recall curves on the new test dataset (BP)

RESULTS

Table 5. The results of the nine methods on human proteins in the new test dataset.[a]

Method	F_{max}			AUPR		
	CC	MF	BP	CC	MF	BP
Naïve	0.620	0.292	0.28	0.538	0.135	0.163
Diamond	0.509	0.516	0.445	0.085	0.087	0.055
DeepGOCNN	0.648	0.419	0.363	0.636	0.253	0.245
TALE	0.675	0.406	0.367	**0.714**	0.324	0.279
DeepFRI	0.561	0.352	0.394	0.431	0.162	0.203
TransFun	**0.686**	**0.538**	**0.468**	0.694	**0.471**	**0.445**
DeepGOPlus	0.657	0.554	0.523	0.631	0.417	0.366
TALE+	**0.689**	0.569	0.497	**0.724**	0.539	0.415
TransFun+	0.684	**0.612**	**0.553**	0.719	**0.557**	**0.499**

[a] Green denotes the individual methods and blue the composite methods. The best results in each type of methods are highlighted bold.

- Performance on human proteins

RESULTS

Table 6. The results of the nine methods on mouse proteins in the new test dataset.[a]

Method	F_{max}			AUPR		
	CC	MF	BP	CC	MF	BP
Naive	0.503	0.235	0.280	0.333	0.100	0.163
Diamond	0.471	0.569	0.379	0.087	0.119	0.082
DeepGOCNN	0.522	0.430	0.333	0.434	0.272	0.195
TALE	0.519	0.564	0.298	0.502	0.518	0.198
DeepFRI	0.411	0.404	0.282	0.247	0.277	0.154
TransFun	**0.558**	**0.576**	**0.355**	**0.517**	**0.532**	**0.289**
DeepGOPlus	**0.559**	0.615	**0.427**	0.472	0.535	0.286
TALE+	0.533	**0.625**	0.408	0.516	0.596	0.293
TransFun+	0.557	0.624	0.403	**0.529**	**0.618**	**0.352**

[a] Green denotes the individual methods and blue the composite methods. The best results in each type of methods are highlighted bold.

- Performance on mouse proteins

RESULTS

Table 7. The results on proteins longer than 1022 residues in the new test dataset.[a]

Method	F_{max}			AUPR		
	CC	MF	BP	CC	MF	BP
Naive	0.493	0.305	0.299	0.331	0.115	0.184
Diamond	0.560	0.583	0.427	0.060	0.093	0.086
DeepGOCNN	**0.551**	0.478	0.329	0.478	0.299	0.209
TALE	0.500	0.435	0.297	0.456	0.349	0.185
DeepFRI	0.483	0.320	0.312	0.277	0.166	0.120
TransFun	0.550	**0.556**	**0.399**	**0.525**	**0.492**	**0.353**
DeepGOPlus	**0.602**	0.622	0.436	0.544	0.558	0.333
TALE+	0.549	**0.675**	0.407	0.498	**0.614**	0.305
TransFun+	0.564	0.593	**0.443**	**0.563**	0.610	**0.396**

[a] Green denotes the individual methods and blue the composite methods. The best results in each type of methods are highlighted bold.

- Performance on proteins longer than 1022 residues

CONCLUSION

- TransFun predicts protein functions from AlphaFold models and protein sequences.
- TransFun uses transfer learning with a protein language model (ESM-1b) to extract sequence features.
- TransFun uses a graph representation to store structural features generated from AlphaFold predicted structures.
- TransFun uses EGNN to predict GO terms.

- Future work:
  - use the multiple sequence alignment (MSA) of a target protein for the MSA-based language model (e.g. ESM-MSA) to generate extra embedding features
  - incorporate the protein–protein interaction information