



Article

GTPLM-GO: Enhancing Protein Function Prediction Through Dual-Branch Graph Transformer and Protein Language Model Fusing Sequence and Local–Global PPI Information

Haotian Zhang ¹, Yundong Sun ^{1,2} , Yansong Wang ¹ , Xiaoling Luo ³ , Yumeng Liu ⁴ , Bin Chen ¹,
Xiaopeng Jin ^{4,*} and Dongjie Zhu ^{1,*}

¹ School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264209, China; zhanght282018@163.com (H.Z.); hitffmy@163.com (Y.S.); yansongwang_hit@stu.hit.edu.cn (Y.W.); bchen@hit.edu.cn (B.C.)

² Department of Electronic Science and Technology, Harbin Institute of Technology, Harbin 150001, China

³ College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China; xiaolingluo@outlook.com

⁴ College of Big Data and Internet, Shenzhen Technology University, Shenzhen 518118, China; liuyumeng@sztu.edu.cn

* Correspondence: jinxiaopeng@sztu.edu.cn (X.J.); zhudongjie@hit.edu.cn (D.Z.)

Abstract: Currently, protein–protein interaction (PPI) networks have become an essential data source for protein function prediction. However, methods utilizing graph neural networks (GNNs) face significant challenges in modeling PPI networks. A primary issue is over-smoothing, which occurs when multiple GNN layers are stacked to capture global information. This architectural limitation inherently impairs the integration of local and global information within PPI networks, thereby limiting the accuracy of protein function prediction. To effectively utilize information within PPI networks, we propose GTPLM-GO, a protein function prediction method based on a dual-branch Graph Transformer and protein language model. The dual-branch Graph Transformer achieves the collaborative modeling of local and global information in PPI networks through two branches: a graph neural network and a linear attention-based Transformer encoder. GTPLM-GO integrates local–global PPI information with the functional semantic encoding constructed by the protein language model, overcoming the issue of inadequate information extraction in existing methods. Experimental results demonstrate that GTPLM-GO outperforms advanced network-based and sequence-based methods on PPI network datasets of varying scales.

Keywords: protein function prediction; PPI networks; dual-branch graph transformer; graph neural networks; protein language model



Academic Editor: Ekaterina Poverennaya

Received: 16 March 2025

Revised: 21 April 2025

Accepted: 23 April 2025

Published: 25 April 2025

Citation: Zhang, H.; Sun, Y.; Wang, Y.; Luo, X.; Liu, Y.; Chen, B.; Jin, X.; Zhu, D. GTPLM-GO: Enhancing Protein Function Prediction Through Dual-Branch Graph Transformer and Protein Language Model Fusing Sequence and Local–Global PPI Information. *Int. J. Mol. Sci.* **2025**, *26*, 4088. <https://doi.org/10.3390/ijms26094088>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Proteins are fundamental to biological processes, and accurate resolution of protein functions is crucial for studies such as revealing disease mechanisms and discovering drug targets [1]. Protein functions are standardized by Gene Ontology (GO) [2] through a structured, controlled vocabulary, which involves three aspects: biological process ontology (BPO), molecular function ontology (MFO), and cellular component ontology (CCO). With advancements in sequencing technologies, protein sequence data are increasing rapidly [3]. At present, the number of protein sequences in UniProt has surpassed 250 million [4]. However, due to expensive and time-consuming biochemical experiments, only approximately 0.23% of these sequences have reliable functional annotations [5]. This indicates

that biochemical experiments are insufficient to meet the growing demand for protein function annotation. Consequently, it is imperative to develop protein function prediction methods [6,7].

Various data sources can be used for protein function prediction. Sequence data are especially prevalent due to their abundance, which has driven significant technological evolution of sequence-based methods. Classical sequence-based methods, such as Blast [8], PSI-Blast [9], and Blast2GO [10], typically transfer annotations between homologous proteins by sequence similarity comparison [6,11]. The development of deep learning has led researchers to incorporate neural network models for sequence analysis. For instance, DeepGO [12] and DeepGOPlus [13] utilize convolutional neural networks to extract feature information from protein sequences, while TALE [14] is based on Transformer [15], which captures global patterns in sequences. Recent developments have also seen the integration of natural language processing (NLP) techniques, with the introduction of protein language models (pLMs) such as SeqVec [16], ProtTrans [17], and ESM-1b [18]. These models, built upon pre-trained models like ELMo [19] and BERT [20], leverage large-scale unlabeled sequences for pre-training and provide high-quality functional semantic encoding for protein function prediction.

In general, the value of sequence data lies in two aspects: (1) providing vast sequence resources that support data-driven research and (2) offering key biological features, such as conserved structural domains and functional sites, which are essential for analyzing the biochemical properties of proteins. To improve the comprehensiveness and accuracy of protein function prediction, it is essential to examine the functional roles of proteins within biological systems [21,22]. Protein–protein interaction (PPI) networks represent proteins as nodes and characterize their interactions through weighted edges [23]. This network topology can effectively reveal functional collaboration among proteins. Therefore, PPI networks are also crucial for protein function prediction [22,24].

Early studies primarily used unsupervised network embedding methods [25] for PPI network information extraction. For instance, DeepGO [12] constructs a heterogeneous network based on PPI networks and generates embeddings of proteins using the DeepWalk algorithm [26] and skip-gram model [27,28]. deepNF [29] integrates multisource PPI networks, constructs a Positive Pointwise Mutual Information (PPMI) matrix via a random wandering strategy, and learns low-dimensional representations through an autoencoder. Recently, GNNs [30] have proven to be highly effective in biological networks. For example, sAMPpred-GAT [31] uses a Graph Attention Network (GAT) to capture residue-level features and structural information from predicted peptide structures to predict antimicrobial peptides (AMPs). Additionally, in the field of protein function prediction, several studies have also utilized GNNs. DeepGraphGO [32] utilizes a two-layer graph convolutional network (GCN) [33] for feature learning across PPI networks, and it is one of the most advanced network-based methods. HNetGO [34] employs a graph neural network based on attention mechanisms for protein embedding learning. Notably, although existing network-based methods effectively capture local features of PPI networks, the performance is constrained by the message passing mechanism. Excessive stacking of graph network layers can lead to the over-smoothing problem [35,36] and ultimately limit the ability to capture global PPI features.

In PPI networks, proteins are connected through direct interactions. However, even in the absence of direct interactions, proteins may still maintain functional relationships through multihop connections, suggesting that distant proteins can indirectly collaborate in biological processes. Therefore, effective collaborative modeling of local–global features in PPI networks is crucial for further enhancing prediction accuracy. In recent years, Transformer [15] has achieved success across various domains, including computer

vision and graph data analysis. Researchers have developed several Graph Transformer models [37–40]. The global attention mechanism of Transformer enables it to capture long-range dependencies and provides novel approaches for PPI network analysis. For example, CFAGO [41] extracts first-order neighborhood features of proteins via a weighted neighborhood matrix of PPI networks, and enhances its ability to model global information through the Transformer encoder. However, CFAGO's extraction of local information is limited to its inputs. To date, existing methods have not yet achieved the collaborative modeling of local and global information of PPI networks. Dual-branch architectures have demonstrated their effectiveness in processing different scales or types of information across various fields. For instance, ViLBERT [42] improves multimodal task performance by simultaneously processing visual and linguistic modalities, while Conformer [43] combines a CNN and Transformer to jointly model both local image details and global context. In protein function prediction, Struct2GO [44] verifies the potential of the dual-branch architecture by integrating sequence and structural features. DeepFMB [45] and DeepFMB+ [45] can be considered as multibranch neural network architectures, which enhance protein function prediction performance by integrating sequences, PPIs, and orthology relations. Building on these concepts, constructing a two-branch graph network for protein–protein interaction (PPI) networks to capture both local and global information may be an effective approach to improving prediction accuracy.

In this study, we propose GTPLM-GO, a protein function prediction method that integrates local and global information within PPI networks, as well as sequence data. Specifically, we developed a dual-branch Graph Transformer to achieve the collaborative modeling of local and global information within the PPI network while also incorporating functional semantic encoding of sequences generated by a protein language model, SeqVec [16]. GTPLM-GO significantly enhances protein function prediction performance. This paper makes the following key contributions:

- We propose a novel protein function prediction method, GTPLM-GO, which effectively utilizes the functional semantic encoding of sequences, as well as local and global information of PPI networks. GTPLM-GO enhances protein function prediction accuracy by leveraging the complementarity of this information.
- We develop a dual-branch Graph Transformer that integrates both local and global information from PPI networks. The local information is extracted using a two-layer GCN branch, while the global information is captured through a linear attention-based Transformer encoder. This design effectively mitigates the over-smoothing problem commonly found in traditional graph networks and achieves collaborative modeling of local and global information.
- Experimental results on PPI network datasets of varying scales show that GTPLM-GO outperforms advanced network-based and sequence-based methods. This confirms its ability to extract valuable information from PPI networks and validate information complementation. Furthermore, GTPLM-GO exhibits notable advantages in species-specific protein function prediction, indicating its good generalization capabilities.

2. Results

2.1. Experimental Setup

GTPLM-GO is trained and evaluated for MF, BP, and CC separately. To optimize the hyperparameters, we used wandb (<https://wandb.ai/site>) as a hyperparameter search tool to prefer a set of optimal hyperparameters based on the valid set loss values. Subsequently, we apply the trained model to the test set to make predictions and evaluate results. In terms of parameter settings, specifically, the batch size is 1024, the dual-branch Graph Transformer hidden dimension size $d = 1024$, the number of layers in the GCN branch = 2,

the number of attention layers in the linear attention-based Transformer encoder = 1, the number of attention headers = 1, and dropout hyperparameters are {0.2, 0.3, 0.4, 0.5}. We use Adam optimizer [46], with learning rate and weight decay values as hyperparameters and the search rates for learning rate in $\{1 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}, 5 \times 10^{-4}\}$ and for weight decay in $\{1 \times 10^{-5}, 1 \times 10^{-6}, 0\}$, respectively. Additionally, we use an early-stopping strategy, limiting the maximum number of training epochs to 1000, and the Patience value is 15, meaning training is stopped early if the valid set loss no longer decreases within 15 epochs. We trained the benchmark methods following the original authors' implementations. As a network-based method, GTPLM-GO uses only those training samples that appear in both the train set and the PPI network, which is consistent with DeepGraphGO. Other benchmark methods utilize all training samples.

2.2. Evaluation Metrics

We used Fmax and AUPR as evaluation metrics, which are widely used to evaluate the performance of protein function prediction [11,32,41,44,47]. AUPR represents the area under the Precision–Recall curve (P–R curve), which is widely used in unbalanced datasets, as well as multilabel classification tasks [11,32]. Fmax is the primary evaluation metric of CAFA [6,7], and can be defined as follows:

$$F_{max} = \max_{\tau \in [0,1]} \left\{ \frac{2 \times pr(\tau) \times rc(\tau)}{pr(\tau) + rc(\tau)} \right\} \quad (1)$$

where $pr(\tau)$ and $rc(\tau)$ represent precision and recall computed at the threshold τ , which can be defined as

$$pr(\tau) = \frac{1}{N(\tau)} \sum_{i=1}^{N(\tau)} \frac{\sum_j s(p_{ij} \geq \tau \text{ and } f_j \in F(i))}{\sum_j s(p_{ij} \geq \tau)} \quad (2)$$

$$rc(\tau) = \frac{1}{N_t(\tau)} \sum_{i=1}^{N_t(\tau)} \frac{\sum_j s(p_{ij} \geq \tau \text{ and } f_j \in F(i))}{\sum_j s(f_j \in F(i))} \quad (3)$$

where $N(\tau)$ denotes the number of proteins with a predicted score of at least one GO term that is not less than the threshold τ , $N_t(\tau)$ denotes the number of proteins in the test set, $s(\cdot)$ is the function used to convert the boolean value “true” to 1 and “false” to 0, f_j represent GO term j , and $F(i)$ denotes the set of GO terms associated with protein $prot_i$.

2.3. Comparison with Advanced Methods on PPI Networks of Varying Scales

As shown in Table 1, we use Fmax and AUPR as metrics to compare GTPLM-GO with six benchmark methods on the benchmark dataset. For details on benchmark methods, see Supplementary Materials. The benchmark dataset contains a large-scale PPI network constructed based on a multispecies strategy. Experimental results demonstrate that GTPLM-GO surpasses all competing methods in both Fmax and AUPR.

Specifically, GTPLM-GO outperforms the advanced network-based method DeepGraphGO in all metrics. In terms of Fmax, GTPLM-GO achieves performance improvements of 2.1%, 2.9%, and 1.3% in BPO, MFO, and CCO, respectively, compared with DeepGraphGO. For AUPR, the improvements are 11.3%, 4.4%, and 4.3%, respectively. Results show that GTPLM-GO has excellent performance on large-scale datasets. Furthermore, both GTPLM-GO and DeepGraphGO, as models that incorporate sequence and PPI network, outperform the four sequence-based methods. Compared with the best results of the sequence-based methods across six metrics, GTPLM-GO improves Fmax by 15.2% (BPO, DeepGOPlus), 3.9% (MFO, LR-InterPro), and 4.3% (CCO, DeepGOPlus). For AUPR, GTPLM-GO shows improvements of 50.0% (BPO, LR-InterPro), 7.0% (MFO, LR-InterPro), and 42.5% (CCO, LR-InterPro). These results suggest that GTPLM-GO enhances prediction

accuracy by combining the functional semantic coding extracted from the protein language model with local and global information within the PPI network.

Table 1. Performance comparison of GTPLM-GO and advanced methods on benchmark dataset *. **Bold** indicates the best performance, and underline indicates the second best.

Method	Fmax			AUPR		
	MFO	BPO	CCO	MFO	BPO	CCO
BLAST-KNN ¹ [48]	0.590	0.274	0.650	0.455	0.113	0.570
LR-InterPro ¹ [48]	0.617	0.278	0.661	0.530	0.144	0.672
Net-KNN ¹ [49]	0.426	0.305	0.667	0.276	0.157	0.641
DeepGOCNN ¹ [13]	0.434	0.248	0.632	0.306	0.101	0.573
DeepGOPlus ² [13]	0.593	0.290	0.672	0.398	0.108	0.595
DeepGraphGO ¹ [32]	<u>0.623</u>	<u>0.327</u>	<u>0.692</u>	<u>0.543</u>	<u>0.194</u>	<u>0.695</u>
GTPLM-GO ²	0.641	0.334	0.701	0.567	0.216	0.725

* Benchmark dataset refers to the dataset proposed by DeepGraphGO. ¹ Single algorithm. ² Composite algorithm.

Notably, the performance of all methods on BPO is significantly lower than that of MFO and CCO. This discrepancy may stem from the complexity of biological processes involved in BPO, which typically require interactions between multiple proteins and network regulation. Additionally, BPO contains a large number of GO terms, which increases the prediction difficulty. However, the performance improvement of GTPLM-GO on BPO is significant, achieving an 11.3% increase in AUPR. This is probably because proteins connected within PPI networks through both direct (physical) and indirect (functional) interactions are typically involved in common biological processes. GTPLM-GO's dual-branch Graph Transformer efficiently models the PPI network by incorporating both local and global information, resulting in enhanced performance.

To further evaluate the performance and adaptability of GTPLM-GO on PPI networks of varying scales, we employed the dataset proposed by CFAGO [41]. Details about CFAGO and its proposed dataset can be found in the Supplementary Materials (Table S1). The CFAGO dataset contains two subsets: HUMAN and MOUSE, each containing a small-scale, single-species PPI network. We retrained GTPLM-GO and compared its performance against DeepGraphGO and CFAGO. As shown in Figure 1, GTPLM-GO consistently maintains its superior performance. Specifically, on the human dataset, GTPLM-GO surpasses DeepGraphGO (0.327, 0.142, 0.209) and CFAGO (0.439, 0.236, 0.366), achieving an Fmax of 0.494, 0.329, and 0.412 for BPO, MFO, and CCO, respectively. Similarly, on the mouse dataset, GTPLM-GO attains an Fmax of 0.284, 0.626, and 0.495 for BPO, MFO, and CCO, respectively, outperforming DeepGraphGO (0.177, 0.321, 0.312) and CFAGO (0.273, 0.514, 0.465). These results indicate that GTPLM-GO excels even in small-scale PPI networks, highlighting its strong adaptability.

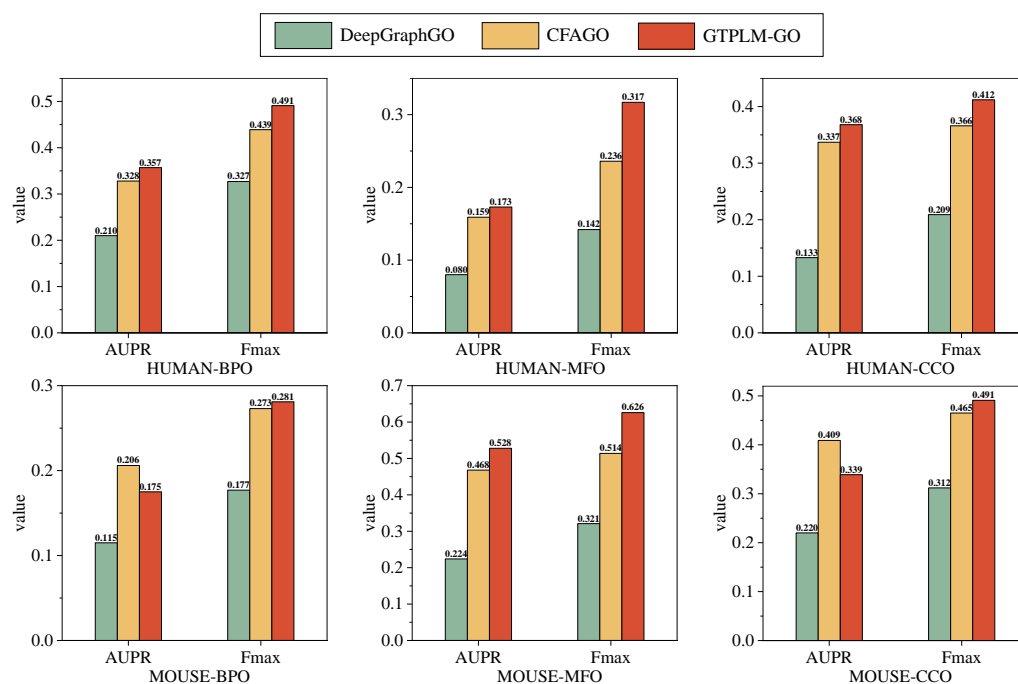


Figure 1. Performance comparison of GTPLM-GO with advanced methods on the CFAGO dataset.

2.4. Generalization on Proteins Within the PPI Network

The protein samples present in both the PPI network and test set were selected as a subset to further evaluate the ability of GTPLM-GO to extract information from the PPI network. The experimental results indicate that GTPLM-GO effectively utilizes PPI network data to improve protein function prediction performance.

Following the experimental setup of DeepGraphGO [32], we constructed two test subsets based on the presence or absence of protein samples in the PPI network, using the test set from the benchmark dataset. The two subsets are “STRING” and “HOMO”. STRING refers to proteins that are present in both test set and the PPI network; HOMO denotes proteins absent from the PPI network but homologous to those within PPI network; and NONE includes all other proteins. The number of STRING proteins in the MFO, BPO, and CCO are 286, 638, and 446, respectively, while the number of HOMO proteins are 132, 246, and 756. We then compared the performance of GTPLM-GO with advanced methods in STRING, and HOMO subsets. As shown in Table 2, GTPLM-GO outperforms other competing methods in 10 out of 12 metrics, with only a slight difference from DeepGraphGO in CCO of STRING (Fmax: −1.5%, AUPR: −0.2%). GTPLM-GO demonstrates superior performance in predicting protein functions in both STRING and HOMO. For instance, in the BP domain, GTPLM-GO (AUPR: 0.245) improves AUPR by 17.2% over DeepGraphGO (AUPR: 0.209) in STRING. Similarly, GTPLM-GO (AUPR: 0.192) outperforms DeepGraphGO (AUPR: 0.157) by 22.3% in HOMO. DeepGraphGO can be considered as the second best method in both STRING and HOMO subsets. Overall, network-based methods significantly outperform sequence-based methods, likely because they can utilize PPI information to predict protein functions, enabling the model to capture more functional associations among proteins.

Table 2. Performance comparison of GTPLM-GO with advanced methods on STRING and homologous proteins on the benchmark dataset *. **Bold** indicates the best performance, and underline indicates the second best.

Method	AUPR			Fmax		
	MFO	BPO	CCO	MFO	BPO	CCO
STRING						
BLAST-KNN ¹ [48]	0.466	0.122	0.438	0.608	0.291	0.570
LR-InterPro ¹ [48]	0.562	0.162	0.598	0.630	0.293	0.627
Net-KNN ¹ [49]	0.297	0.177	0.607	0.443	0.314	0.617
DeepGOCNN ¹ [13]	0.173	0.036	0.136	0.432	0.258	0.588
DeepGOPlus ² [13]	0.423	0.118	0.489	0.602	0.306	0.617
DeepGraphGO ¹ [32]	<u>0.582</u>	<u>0.209</u>	0.663	<u>0.642</u>	<u>0.348</u>	0.665
GTPLM-GO ²	0.607	0.245	<u>0.662</u>	0.653	0.350	<u>0.655</u>
HOMO						
BLAST-KNN ¹ [48]	0.456	0.104	0.652	0.583	0.248	0.704
LR-InterPro ¹ [48]	<u>0.501</u>	0.114	0.720	0.602	0.256	0.689
Net-KNN ¹ [49]	0.253	0.128	0.675	0.422	0.300	0.709
DeepGOCNN ¹ [13]	0.349	0.088	0.613	0.456	0.231	0.662
DeepGOPlus ² [13]	0.438	0.100	0.656	0.582	0.257	0.710
DeepGraphGO ¹ [32]	0.475	<u>0.157</u>	<u>0.736</u>	<u>0.619</u>	<u>0.306</u>	<u>0.726</u>
GTPLM-GO ²	0.517	0.192	0.781	0.641	0.323	0.738

* Benchmark dataset refers to the dataset proposed by DeepGraphGO. ¹ Single algorithm. ² Composite algorithm.

2.5. Generalization on Specific Species

To evaluate the generalization capability of GTPLM-GO on species-specific proteins, we compared GTPLM-GO with advanced methods on the HUMAN and MOUSE test subsets. The benchmark dataset as well as the PPI network were constructed based on a multispecies strategy [32]. This strategy enabled the model to integrate multispecies PPI data during training. Then, the model predicts protein functions across various species. Therefore, to assess the generalization ability of GTPLM-GO on specific species, we first evaluated its performance on the species-specific test subsets of benchmark dataset. We use HUMAN and MOUSE proteins listed in Table 6 as instances, which is consistent with DeepGraphGO. Table 3 presents the comparison results of GTPLM-GO and advanced methods. GTPLM-GO achieved the best performance in 10 of the 12 metrics. Specifically, GTPLM-GO achieved optimal performances on the MOUSE test set in all metrics, while in the human test set, it only underperformed in MF. This suggests that GTPLM-GO trained using a multispecies strategy is more effective at extracting common features of proteins from multispecies PPI data.

Furthermore, we also investigated the ability of GTPLM-GO and DeepGraphGO to generalize to the HUMAN and MOUSE test set proteins when trained using only the target species proteins. This actually resulted in two variants, GTPLM-GO_{species} and DeepGraphGO_{species}, respectively. Experimental results can be found in the Supplementary Materials (Table S2). Specifically, the performance of both variants generally declined when trained only on target species proteins. However, GTPLM-GO still outperforms DeepGraphGO on several domains. For example, in terms of AUPR, GTPLM-GO_{HUMAN} and GTPLM-GO_{MOUSE} achieve 0.764 and 0.641 on CCO, respectively, outperforming DeepGraphGO (0.642, 0.634). This further demonstrates the generalization of GTPLM-GO on specific species.

Table 3. Performance comparison of GTPLM-GO with advanced methods on HUMAN and MOUSE proteins on the benchmark dataset *. **Bold** indicates the best performance, and underline indicates the second best.

Method	AUPR			Fmax		
	MFO	BPO	CCO	MFO	BPO	CCO
HUMAN						
BLAST-KNN ¹ [48]	0.296	0.074	0.384	0.471	0.241	0.555
LR-InterPro ¹ [48]	<u>0.496</u>	0.138	0.603	<u>0.593</u>	0.282	0.650
Net-KNN ¹ [49]	0.358	0.143	0.620	0.485	0.261	0.615
DeepGOCNN ¹ [13]	0.327	0.114	0.552	0.468	0.263	0.594
DeepGOPlus ² [13]	0.246	0.088	0.479	0.501	0.277	0.625
DeepGraphGO ¹ [32]	0.520	<u>0.178</u>	<u>0.642</u>	0.633	<u>0.320</u>	<u>0.655</u>
GTPLM-GO ²	0.471	0.185	0.777	0.588	0.327	0.732
MOUSE						
BLAST-KNN ¹ [48]	0.593	0.105	0.441	<u>0.681</u>	0.289	0.593
LR-InterPro ¹ [48]	0.625	0.175	0.569	0.628	0.312	0.592
Net-KNN ¹ [49]	0.319	0.167	0.569	0.420	0.302	0.588
DeepGOCNN ¹ [13]	0.405	0.129	0.495	0.475	0.258	0.574
DeepGOPlus ² [13]	0.550	0.132	0.488	0.634	0.306	0.598
DeepGraphGO ¹ [32]	<u>0.651</u>	<u>0.201</u>	<u>0.634</u>	0.650	<u>0.329</u>	<u>0.638</u>
GTPLM-GO ²	0.653	0.203	0.679	0.701	0.334	0.682

* Benchmark dataset refers to the dataset proposed by DeepGraphGO. ¹ Single algorithm. ² Composite algorithm.

2.6. Ablation Studies

We performed ablation experiments to evaluate the contribution of the dual-branch Graph Transformer and protein language model. One component was removed at a time, and the model was retrained under the same experimental setup to assess the impact of each component on performance. Specifically, two variants were created: GTPLM-GO_{w/o ppi}, which retains only the functional semantic encoding of sequences generated by SeqVec to construct sequence-based embeddings of proteins, and GTPLM-GO_{w/o seq}, which retains only the dual-branch Graph Transformer to construct PPI-based embeddings. The detailed performance comparison of these variants with DeepGraphGO and GTPLM-GO is shown in Table 4. The results demonstrate that removing any component leads to a performance decline in GTPLM-GO, thus confirming that each component contributes positively to protein function prediction.

Table 4. Ablation study of dual-branch Graph Transformer and protein language model used by GTPLM-GO on the benchmark dataset. **Bold** indicates the best performance, and underline indicates the second best.

Method	Fmax			AUPR		
	MFO	BPO	CCO	MFO	BPO	CCO
GTPLM-GO _{w/o ppi}	0.573	0.263	0.676	0.485	0.139	0.689
GTPLM-GO _{w/o seq}	<u>0.637</u>	<u>0.324</u>	<u>0.694</u>	<u>0.552</u>	<u>0.211</u>	<u>0.721</u>
GTPLM-GO	0.641	0.334	0.701	0.567	0.216	0.725

Meanwhile, as shown in Table 4, the removal of the dual-branch Graph Transformer results in a significant performance drop (GTPLM-GO_{w/o ppi}), emphasizing the crucial role of local–global PPI information in protein function prediction. Additionally, as shown in Figure 2, we validate the contribution of dual-branch Graph Transformer in extracting PPI network information by comparing the performance of GTPLM-GO_{w/o seq}, DeepGraphGO, and GTPLM-GO on the benchmark dataset. The results indicate that GTPLM-GO_{w/o seq}

performs the second-best in five out of six performance metrics, which suggests that the dual-branch Graph Transformer significantly outperforms DeepGraphGO in PPI network modeling. Simultaneously, the results also imply that the Graph Transformer has potential in the field of PPI network information mining and protein function prediction. Furthermore, GTPLM-GO achieves the best performance across all six metrics, highlighting that the integration of PPI network embedding with functional semantic coding of sequences constructed by the protein language model helps to enhance protein function prediction accuracy. It is also shown that the protein language model SeqVec is capable of extracting functional features that are not captured by the PPI network or InterPro features. We further refined the experimental setup for the ablation studies to assess the impact of removing the linear attention-based Transformer encoder and GCN on model performance. The specific experimental results can be found in the Supplementary Materials (Table S3).

Subsequently, the impact of utilizing different protein language models (pLMs) on the prediction performance of GTPLM-GO was investigated. Specifically, we replaced the SeqVec model used in GTPLM-GO with four advanced pLMs: ProtBert [17], ESM-1b [18], ESM2 [50], and ProtT5 [17], resulting in four distinct variants. As shown in Table 5, SeqVec has the smallest number of parameters, which is 93 M. The parameter sizes of the ProtBert, ESM-1b, and ESM2 models we used are similar, at 420 M, 650 M, and 650 M, respectively. ProtT5 has the largest number of trainable parameters among the selected pLMs. These variants were retrained on the benchmark dataset, and their performance in protein function prediction was evaluated. As shown in Figure 3, GTPLM-GO achieved the best performance in MFO. GTPLM-GO_{ProtT5} and GTPLM-GO_{ESM-1b} attained the best and second-best performance in BPO, respectively. This may be due to the fact that the BPO task is more challenging on benchmark datasets. ESM-1b and ProtT5, with their larger parameter scales and more complex models, are better able to extract more information from the sequence, thereby improving performance on the BPO task. In CCO, GTPLM-GO_{ProtT5} performed well, achieving the best Fmax and the second-best AUPR. GTPLM-GO_{ProtBert} also performed well in CCO. Overall, both GTPLM-GO and its variants outperformed DeepGraphGO in all cases, suggesting that pLMs effectively capture knowledge relevant to functional prediction. Notably, although more advanced pLMs with more parameters, such as ESM-2 (650 M parameters) and ProtT5 (3B parameters), were used to construct variants of GTPLM-GO, no significant improvement was observed. However, this does not imply that larger-scale pLMs cannot yield potential performance gains. The limited performance differences suggest that the current GTPLM-GO architecture does not fully exploit the deep semantic information learned by pLMs. Therefore, future work will focus on optimizing the architectural design to better leverage the advantages of pLMs and enhance the accuracy of protein function prediction. Within the GTPLM-GO framework, the choice of different pLMs appears to have a relatively minor effect on performance. Given the relatively small number of trainable parameters in SeqVec (93M), we selected SeqVec to generate sequence feature semantic encodings in GTPLM-GO, prioritizing efficiency. Additionally, PPI data typically originate from prior knowledge, computational interaction predictions, and experimental data [51], often containing noise, which may be an important reason limiting the further improvement of the prediction accuracy of GTPLM-GO. We further performed hyperparameter studies, which can be found in the Supplementary Materials (Tables S4–S6, Figure S1).

Table 5. Detailed statistics of protein language models (pLMs) used in different GTPLM-GO variants.

Variant	Protein Language Model (pLM)	Parameters
GTPLM-GO	SeqVec	93 M
GTPLM-GO _{ProtBert}	ProtBert	420 M
GTPLM-GO _{ESM-1b}	esm1b_t33_650M_UR50S	650 M
GTPLM-GO _{ESM-2}	esm2_t33_650M_UR50D	650 M
GTPLM-GO _{ProtT5}	ProtT5-XL-UniRef50	3B

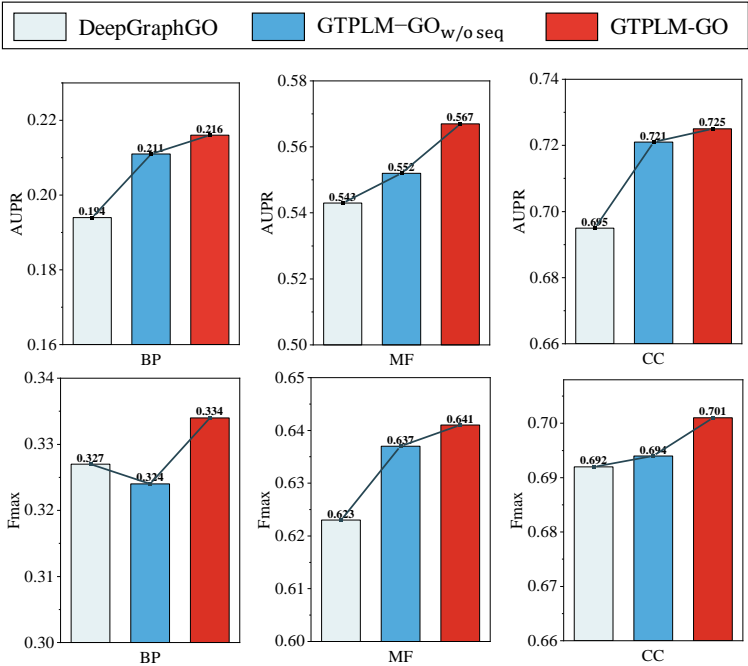


Figure 2. Performance comparison of GTPLM-GO, GTPLM-GO_{w/o seq}, and DeepGraphGO on the benchmark dataset.

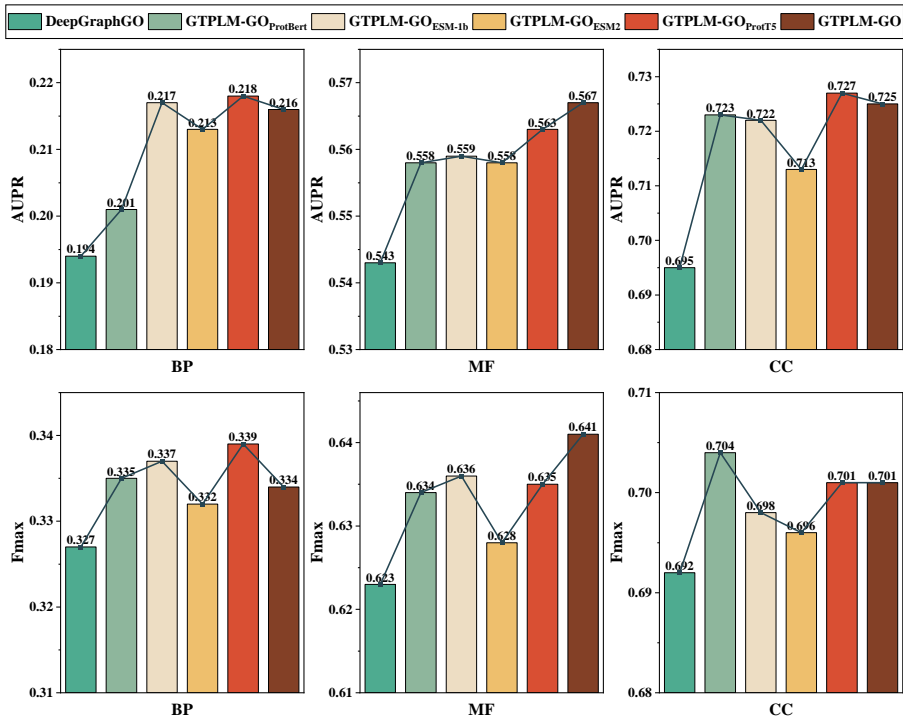


Figure 3. Performance comparison of GTPLM-GO variants using different protein language models on the benchmark dataset.

3. Materials and Methods

3.1. Datasets

In this study, we utilize the benchmark dataset proposed by DeepGraphGO [32], which is collected following the standard Critical Assessment of Function Annotation (CAFA) protocol [7]. Specifically, the PPI network is constructed using a multispecies strategy, downloaded from STRING v11.0 [52], and protein sequences are obtained from UniProt [4]. GO terms are obtained from SwissProt [3], GOA [53], and GO [2]. Only experimental annotations (GO terms) with evidence codes: IDA, IPI, EXP, IGI, IMP, IEP, IC, or TA are retained. Protein function prediction can be regarded as a large-scale multilabel, multiclassification problem [12,32,44]. Three ontology domains, MFO, BPO, and CCO, in this dataset contain 6640, 21,288 and 2729 GO terms, respectively, corresponding to the number of labels to be predicted. The dataset is divided in accordance with DeepGraphGO and follows the CAFA protocol, where protein samples from multiple species are divided into train, valid, and test sets based on the timestamp of when the proteins were annotated. Table 6 shows the detailed statistics of the train, valid, and test sets, as well as the information of the subdatasets used in the extended experiments.

Table 6. Detailed statistics of benchmark dataset for three ontology domains: MFO, BPO, and CCO.

	Train			Valid			Test		
	MFO	BPO	CCO	MFO	BPO	CCO	MFO	BPO	CCO
All Data	51,549	85,104	76,098	490	1570	923	426	925	1224
Data used by GTPLM-GO	35,092	54,276	48,093	490	1570	923	426	925	1224
HUMAN (9606)	9208	12,095	18,842	86	138	137	41	87	767
MOUSE (10090)	6138	9927	8482	103	299	228	65	156	130

3.2. Methods

3.2.1. Overview

As shown in Figure 4, GTPLM-GO develops a dual-branch Graph Transformer to enable the collaborative modeling of both local and global information within PPI networks. Subsequently, the PPI features extracted by the dual-branch Graph Transformer are combined with the functional semantic encoding of sequences generated by SeqVec. GTPLM-GO accepts two inputs: (1) PPI network consisting of N proteins (denoted as G). PPI network is represented by the weighted adjacency matrix $A \in \mathbb{R}^{N \times N}$, where the element $a_{i,j}$ in the matrix denotes the confidence of an interaction between proteins $prot_i$ and $prot_j$ if the edge (i, j) exists, and 0 otherwise; (2) protein sequences.

For each protein $prot_i$ in the dataset, GTPLM-GO utilizes the dual-branch Graph Transformer to generate its PPI-based embedding. Specifically, a two-layer GCN branch is employed to extract features based on the local neighborhood and topological information of $prot_i$. Meanwhile, the global PPI functional features of $prot_i$ are captured using a linear attention-based Transformer encoder, with the resulting PPI-based embeddings derived through feature concatenation. Additionally, GTPLM-GO generates functional semantic encoding of a sequence via a protein language model and constructs the sequence-based embedding using an MLP. Finally, GTPLM-GO combines the PPI-based and sequence-based embeddings of $prot_i$ to obtain its final embedding and predicts the scores of $prot_i$ for each GO term through the GO term classifier.

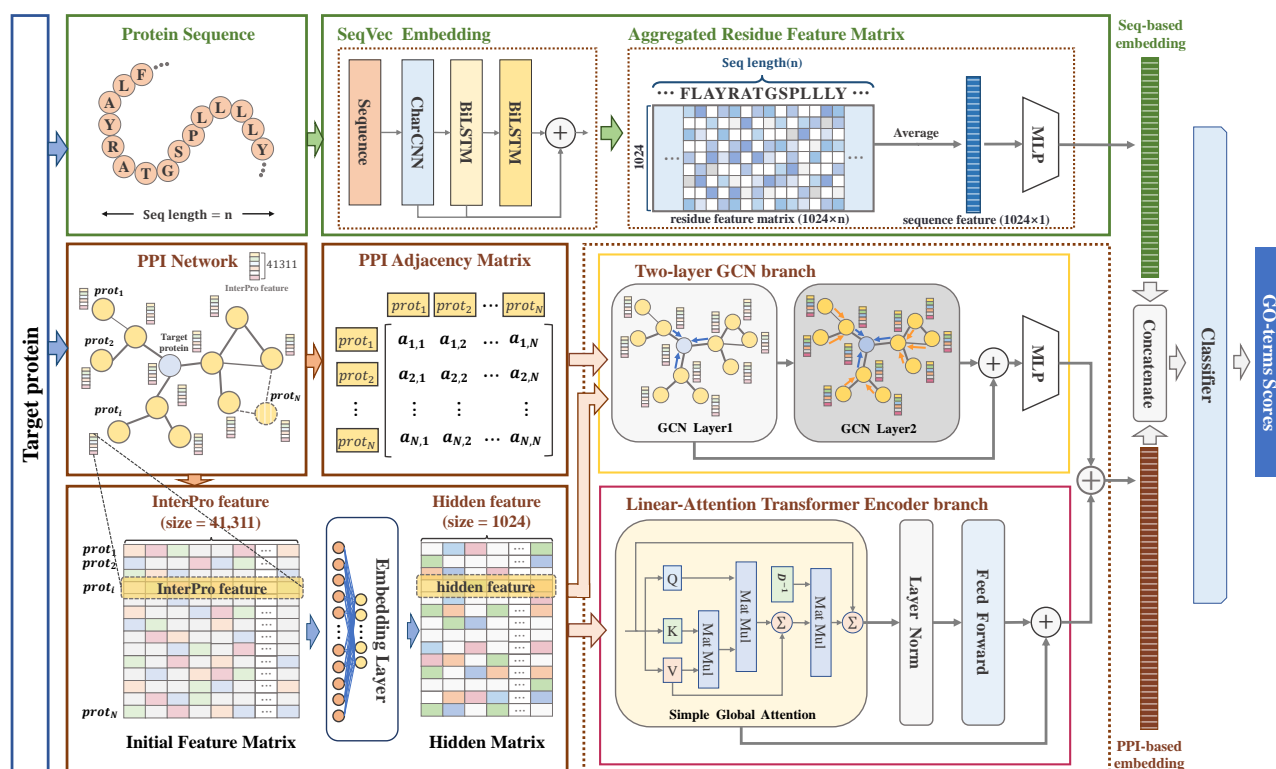


Figure 4. Overall architecture of GTPLM-GO: This model fuses sequence, local, and global PPI information to achieve protein function prediction. It develops a dual-branch Graph Transformer to collaboratively model both local and global information within PPI network. The Graph Transformer takes a PPI adjacency matrix and InterPro feature matrix as input. InterPro feature matrix is constructed by concatenating InterPro features [54] of proteins in the PPI network, while the PPI adjacency matrix encodes both direct and indirect protein interactions. The Graph Transformer extracts local PPI features through a two-layer GCN branch and captures global PPI features using a linear attention-based Transformer encoder. These features are then integrated via feature concatenation to generate PPI-based embeddings. Additionally, GTPLM-GO generates functional semantic representations of sequences using a protein language model, SeqVec [16] and constructs sequence-based embeddings through a multilayer perceptron (MLP). Finally, GTPLM-GO fuses sequence and PPI embeddings as inputs to the classifier to predict GO term scores.

3.2.2. Linear Attention-Based Transformer Encoder for Extracting Global PPI Information

The linear attention-based Transformer encoder is a branch of the dual-branch Graph Transformer. It takes the PPI network (denoted as G) as input. The network G consists of N proteins. For each protein $prot_i$ in G , GTPLM-GO uses the InterPro feature $x_i \in \mathbb{R}^m$ generated by InterProScan [55] as its initial feature, and m represents the feature dimension size. By concatenating the initial features of N protein nodes, we obtain the initial node feature matrix $X_{init} \in \mathbb{R}^{N \times m}$. We then apply a node embedding layer to map the initial feature matrix X_{init} to a low-dimensional hidden matrix $H^{(0)} \in \mathbb{R}^{N \times d}$, which is computed as follows:

$$H^{(0)} = \sigma(X_{init}W^{(0)} + b^{(0)}) \quad (4)$$

where $W^{(0)} \in \mathbb{R}^{m \times d}$ and $b^{(0)} \in \mathbb{R}^d$ represent the learnable weight matrix and bias, respectively, and σ is the nonlinear activation function.

Due to the constraints of the message-passing mechanism, GNNs face challenges in effectively extracting the global features of PPI networks through stacked graph network layers. SGFormer [40] is a simplified yet high-performance Graph Transformer. Inspired by the Simple Global Attention (SGA) mechanism of SGFormer [40], we propose that a single-layer attention network can effectively propagate protein feature information across the PPI network, enabling the modeling of potential functional associations between any

pair of proteins within the PPI network. Therefore, we construct a Transformer encoder consisting of a single layer of the attention network based on the SGA mechanism proposed by SGFormer [40]. The SGA mechanism is defined as follows:

$$Q = f_Q(H^{(0)}), K = f_K(H^{(0)}), V = f_V(H^{(0)}) \quad (5)$$

$$D = \text{diag}\left(1_N + \frac{1}{N}Q(K^T 1_N)\right), \text{SGA} = D^{-1}\left[V + \frac{1}{N}\tilde{Q}(\tilde{K}^T V)\right] + H^{(0)} \quad (6)$$

where f_Q, f_K, f_V denotes the linear layer used to construct Q, K, V , respectively, 1_N is the N -dimensional all-one vector, and the diag operator diagonalizes the N -dimensional vector into an $N \times N$ matrix. Q and K should be normalized using the Frobenius norm before the attention calculation. SGA eliminates the SoftMax operation, resulting in a linear time complexity of $O(N)$, which enables scalability on large-scale PPI networks. The computational procedure of the linear attention-based Transformer encoder, based on the SGA mechanism, is as follows:

$$H_{\text{Trans}} = \text{MLP}\left(\text{LN}\left(\text{SGA}\left(H^{(0)}\right)\right)\right) + H^{(0)} \quad (7)$$

where H_{Trans} denotes the protein feature matrix constructed by the linear attention-based Transformer encoder branch, which enhances the ability of GTPLM-GO to model global information in PPI networks.

3.2.3. Two-Layer GCN for Extracting Local PPI Information

Another component of the dual-branch Graph Transformer is the two-layer GCN. This branch builds upon the GCN proposed by [33], while preserving the ability to model direct(physical) interactions [52] in PPI networks. Inspired by DeepGraphGO [32], we integrate PPI edge weights and residual link [56] into the GCN, and the $(l + 1)$ -th layer is computed as follows:

$$H_{\text{GCN}}^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H_{\text{GCN}}^{(l)}W^{(l)}\right) + H_{\text{GCN}}^{(l)} \quad (8)$$

where $\tilde{A} = A + I_N$ represents the adjacency matrix of the network G , which includes the self-loop. Here, $I_N \in \mathbb{R}^{N \times N}$ is the identity matrix, and $\tilde{D} \in \mathbb{R}^{N \times N}$ denotes the degree matrix of \tilde{A} . Additionally, $W^{(l)} \in \mathbb{R}^{d \times d}$ represents the learnable weights, σ is the nonlinear activation function, and $H_{\text{GCN}}^{(0)} = H^{(0)}$.

From the node perspective, the feature $h_i^{(l+1)} \in \mathbb{R}^d$ of protein $prot_i$ at layer $l + 1$ can be computed as

$$h_i^{(l+1)} = h_i^{(l)} + \sigma\left(\sum_{j \in N(i) \cup \{i\}} \frac{PPI_{i,j}}{\sqrt{\deg(i)}\sqrt{\deg(j)}} W^{(l)} h_j^{(l)}\right) \quad (9)$$

where $N(i)$ denotes the set of neighboring nodes of protein $prot_i$, and $\deg(i)$ and $\deg(j)$ represent the degrees of node $prot_i$ and its neighboring nodes, respectively. GCN aggregates the neighborhood features of the protein nodes, fully accounting for the direct interactions and local neighborhood information of the PPI network.

Finally, the dual-branch Graph Transformer combines the outputs of the linear attention-based Transformer encoder branch with the two-layer GCN branch to obtain the PPI-based embedding matrix $H_{\text{PPI}} \in \mathbb{R}^{N \times 2d}$, as defined by the following equation:

$$H_{\text{PPI}} = H_{\text{GCN}}^{(2)} \parallel H_{\text{Trans}} \quad (10)$$

The dual-branch graph Transformer of GTPLM-GO then achieves the collaborative modeling of both local and global information in PPI networks through a two-layer GCN and a linear attention-based Transformer encoder.

3.2.4. Generating Functional Semantic Encoding Through Protein Language Model

The dual-branch Graph Transformer generates protein embeddings by considering the global perspective of biological networks. However, a protein's sequence determines its structure and function, and these intrinsic features cannot be fully captured by the topological information of the PPI network alone. GTPLM-GO incorporates the protein language model SeqVec [16] to generate functional semantic encoding of sequences and then model the relationships between sequences and functions.

SeqVec comprises a CharCNN layer and two BiLSTM layers, enabling it to model sequences of arbitrary length. Firstly, CharCNN extracts local patterns from the sequence and generates a fixed-dimensional feature for each amino acid. Subsequently, the BiLSTM layers incorporate contextual information from the sequence. For a sequence $prot_i$ containing n amino acids, the feature $SeqVec_j$ of the j -th amino acid is computed as follows [16]:

$$SeqVec_j = h_j^{CharCNN} + h_j^{LSTM_1} + h_j^{LSTM_2} \quad (11)$$

where $h_j^{CharCNN} \in \mathbb{R}^{1024}$ are the local features extracted by CharCNN, and $h_j^{LSTM_1}$ and $h_j^{LSTM_2}$ are the 1024-dimensional contextual features generated by the two BiLSTM layers, respectively. The 1024-dimensional feature of the j -th amino acid is derived by summing the features from the three SeqVec layers. The n residue-level features are then concatenated into a $1024 \times n$ dimensional feature matrix. Subsequently, the 1024-dimensional protein-level sequence features h_i^{Seq} of protein $prot_i$ are derived by averaging these features. Finally, the sequence-based embedding of protein $prot_i$ is obtained through an MLP:

$$emb_i^{Seq} = f(W^{(1)}h_i^{Seq} + b^{(1)}) \quad (12)$$

where $W^{(1)} \in \mathbb{R}^{d \times 1024}$ and $b^{(1)} \in \mathbb{R}^d$ represent the learnable weight matrix and bias, respectively. In the implementation, SeqVec is pre-trained using UniRef50. To enhance the model's running efficiency and facilitate its adaptation to large-scale datasets, we precompute the sequence encodings for the sequences of the proteins in the PPI network using SeqVec in an offline manner. This preprocessing step significantly reduces the computational overhead during model runtime. GTPLM-GO then combines sequence-based embeddings with PPI-based embeddings for protein function prediction.

It is worth noting that constructing models based on multibranch neural networks has become an effective approach to improve protein function prediction, such as DeepFMB [45], DeepFMB+ [45], SpatialPPIv2 [57], and Struct2GO [44]. These models perform tasks such as protein function prediction and PPI prediction by integrating various types of information, such as sequence, PPIs, orthology relations, and structural information. GTPLM-GO integrates PPI and sequence features to enhance predictive performance. GTPLM-GO utilizes a dual-branch Graph Transformer to achieve collaborative modeling of both local and global information in the PPI network, and extracts sequence features using a protein language model. A comparison between GTPLM-GO and the above methods can be found in the Supplementary Materials (Table S7).

3.2.5. Protein Function Classifier Based on Sequence and Local–Global PPI Information

For protein $prot_i$, its embedding emb_i is obtained by combining the PPI-based embedding emb_i^{PPI} and the sequence-based embedding emb_i^{Seq} , as defined in the following equation:

$$emb_i = emb_i^{PPI} \parallel emb_i^{Seq} \quad (13)$$

where \parallel denotes feature concatenation. Strategies such as weighted fusion and attention mechanisms may offer additional advantages. Concatenation was selected for its simplicity and effectiveness in our experiments. This strategy directly combines features from different sources while preserving the independence of features. Further explorations of integration strategies can be found in the Supplementary Materials (Table S8).

Subsequently, GTPLM-GO calculates the predicted score between proteins and GO terms using a GO term classifier, which consists of a fully connected layer, defined as follows:

$$[\hat{y}_{i1}, \hat{y}_{i2}, \dots, \hat{y}_{iM}]^T = f_{out}(emb_i) \quad (14)$$

where M denotes the number of GO terms, and $\hat{y}_{ij} \in \mathbb{R}^{M \times 1}$ represents the confidence that protein $prot_i$ is predicted to belong to the j -th GO term. The classifier implements a mapping from protein embeddings to GO term scores.

GTPLM-GO employs binary cross-entropy loss as the loss function to optimize the model by minimizing the difference between true labels and predicted scores:

$$BCE_{Loss} = -\frac{1}{BM} \sum_{i=1}^B \sum_{j=1}^M y_{ij} \log(\hat{y}_{ij}) + (1 - y_{ij}) \log(1 - \hat{y}_{ij}) \quad (15)$$

where B denotes the number of proteins in one batch, and $y_{ij} \in \{0, 1\}$ is the ground truth of protein $prot_i$.

4. Conclusions

In this paper, we propose GTPLM-GO, a method for protein function prediction by fusing sequence and local-global PPI information. GTPLM-GO achieves collaborative modeling of local and global information of the PPI network by utilizing a dual-branch Graph Transformer. Meanwhile, GTPLM-GO leverages the protein language model SeqVec to extract the functional semantic information of sequences and generates protein embeddings by combining them with PPI network features. GTPLM-GO then addresses the issue of insufficient information extraction in existing protein function prediction methods. Experimental results on datasets of different sizes demonstrate that GTPLM-GO outperforms advanced network-based and sequence-based methods in protein function prediction.

GTPLM-GO introduces a novel approach for extracting PPI network information, demonstrating the considerable potential of Graph Transformer in analyzing complex biological networks. While GTPLM-GO demonstrates excellent performance in protein function prediction, there is still room for further improvement. In the future, we aim to enhance both the applicability and prediction accuracy of GTPLM-GO while addressing issues within PPI networks, such as false positives and false negatives. Moreover, because a protein's structure determines its function [58,59], highly accurate protein structure prediction techniques, such as AlphaFold2 [60], have driven the development of methods that utilize structural information to predict protein functions. A key focus of our future work will be integrating structural features into GTPLM-GO to further improve prediction accuracy. Based on the PDB files generated by AlphaFold2, we will construct residue contact maps and attempt to build protein structure encoders using GNNs or Graph Transformers to extract structural features. Additionally, we will explore ways to integrate structural features into GTPLM-GO, such as incorporating them as the initial features of protein nodes in the PPI network or as part of the initial features.

Supplementary Materials: The following supporting information can be downloaded at <https://www.mdpi.com/article/10.3390/ijms26094088/s1>. References [13,32,33,41,44,45,48,49,54,57] are cited in the Supplementary Materials.

Author Contributions: Conceptualization, H.Z. and Y.S.; methodology, H.Z., Y.S., and X.J.; software, H.Z., and Y.W.; validation, H.Z., and Y.W.; data curation, H.Z., and Y.W.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z., Y.S., Y.W., X.L., Y.L., B.C., X.J., and D.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China (Grant No. 62302317), the Shenzhen Colleges and Universities Stable Support Program (Grant No. 20220715183602001), the Natural Science Foundation of Guangdong Province (Grant 2025A1515010184), the project of Shenzhen Science and Technology Innovation Committee (Grant JCYJ20240813141424032), the Stable Support Projects for Shenzhen Higher Education Institutions (Grant No. 20231122005530001) and Natural Science Foundation of Top Talent of SZTU (Grant No. GDRC202319).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets and source codes (in Pytorch) of GTPLM-GO are available at <https://github.com/gnahzt28/GTPLM-GO>.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Eisenberg, D.; Marcotte, E.M.; Xenarios, I.; Yeates, T.O. Protein function in the post-genomic era. *Nature* **2000**, *405*, 823–826. [CrossRef] [PubMed]
2. Ashburner, M.; Ball, C.A.; Blake, J.A.; Botstein, D.; Butler, H.; Cherry, J.M.; Davis, A.P.; Dolinski, K.; Dwight, S.S.; Eppig, J.T.; et al. Gene ontology: Tool for the unification of biology. *Nat. Genet.* **2000**, *25*, 25–29. [CrossRef] [PubMed]
3. Boutet, E.; Lieberherr, D.; Tognolli, M.; Schneider, M.; Bairoch, A. UniProtKB/Swiss-Prot: The manually annotated section of the UniProt KnowledgeBase. In *Plant bioinformatics: Methods and protocols*; Humana Press: Totowa, NJ, USA, 2007; pp. 89–112. [CrossRef]
4. The UniProt Consortium. UniProt: The Universal Protein Knowledgebase in 2023. *Nucleic Acids Res.* **2022**, *51*, D523–D531. [CrossRef]
5. Costanzo, M.; VanderSluis, B.; Koch, E.N.; Baryshnikova, A.; Pons, C.; Tan, G.; Wang, W.; Usaj, M.; Hanchard, J.; Lee, S.D.; et al. A global genetic interaction network maps a wiring diagram of cellular function. *Science* **2016**, *353*, aaf1420. [CrossRef]
6. Radivojac, P.; Clark, W.T.; Oron, T.R.; Schnoes, A.M.; Wittkop, T.; Sokolov, A.; Graim, K.; Funk, C.; Verspoor, K.; Ben-Hur, A.; et al. A large-scale evaluation of computational protein function prediction. *Nat. Methods* **2013**, *10*, 221–227. [CrossRef] [PubMed]
7. Zhou, N.; Jiang, Y.; Bergquist, T.R.; Lee, A.J.; Kacsóh, B.Z.; Crocker, A.W.; Lewis, K.A.; Georghiou, G.; Nguyen, H.N.; Hamid, M.N.; et al. The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.* **2019**, *20*, 244. [CrossRef]
8. Altschul, S.F.; Gish, W.; Miller, W.; Myers, E.W.; Lipman, D.J. Basic local alignment search tool. *J. Mol. Biol.* **1990**, *215*, 403–410. [CrossRef]
9. Altschul, S.F.; Madden, T.L.; Schäffer, A.A.; Zhang, J.; Zhang, Z.; Miller, W.; Lipman, D.J. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **1997**, *25*, 3389–3402. [CrossRef]
10. Conesa, A.; Götz, S.; García-Gómez, J.M.; Terol, J.; Talón, M.; Robles, M. Blast2GO: A universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics* **2005**, *21*, 3674–3676. [CrossRef]
11. 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. *Briefings Bioinform.* **2023**, *24*, bbad117. [CrossRef]
12. Kulmanov, M.; Khan, M.A.; Hoehndorf, R. DeepGO: Predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics* **2018**, *34*, 660–668. [CrossRef] [PubMed]
13. Kulmanov, M.; Hoehndorf, R. DeepGOPlus: Improved protein function prediction from sequence. *Bioinformatics* **2020**, *36*, 422–429. [CrossRef]
14. Cao, Y.; Shen, Y. TALE: Transformer-based protein function Annotation with joint sequence–Label Embedding. *Bioinformatics* **2021**, *37*, 2825–2833. [CrossRef]

15. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.
16. Heinzinger, M.; Elnaggar, A.; Wang, Y.; Dallago, C.; Nechaev, D.; Matthes, F.; Rost, B. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinform.* **2019**, *20*, 723. [\[CrossRef\]](#)
17. Elnaggar, A.; Heinzinger, M.; Dallago, C.; Rehawi, G.; Wang, Y.; Jones, L.; Gibbs, T.; Feher, T.; Angerer, C.; Steinegger, M.; et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 7112–7127. [\[CrossRef\]](#)
18. Rives, A.; Meier, J.; Sercu, T.; Goyal, S.; Lin, Z.; Liu, J.; Guo, D.; Ott, M.; Zitnick, C.L.; Ma, J.; et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2016239118. [\[CrossRef\]](#) [\[PubMed\]](#)
19. Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Zettlemoyer, L. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Volume 1 (Long Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 2227–2237. [\[CrossRef\]](#)
20. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019; Volume 1 (Long and Short Papers); Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 4171–4186. [\[CrossRef\]](#)
21. Lin, B.; Luo, X.; Liu, Y.; Jin, X. A comprehensive review and comparison of existing computational methods for protein function prediction. *Briefings Bioinform.* **2024**, *25*, bbae289. [\[CrossRef\]](#) [\[PubMed\]](#)
22. Szklarczyk, D.; Nastou, K.; Koutrouli, M.; Kirsch, R.; Mehryary, F.; Hachilif, R.; Hu, D.; Peluso, M.E.; Huang, Q.; Fang, T.; et al. The STRING database in 2025: Protein networks with directionality of regulation. *Nucleic Acids Res.* **2025**, *53*, D730–D737. [\[CrossRef\]](#)
23. Spirin, V.; Mirny, L.A. Protein complexes and functional modules in molecular networks. *Proc. Natl. Acad. Sci. USA* **2003**, *100*, 12123–12128. [\[CrossRef\]](#)
24. Zhang, A. *Protein Interaction Networks: Computational Analysis*; Cambridge University Press: Cambridge, UK, 2009.
25. Cui, P.; Wang, X.; Pei, J.; Zhu, W. A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 833–852. [\[CrossRef\]](#)
26. Perozzi, B.; Al-Rfou, R.; Skiena, S. Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 701–710. [\[CrossRef\]](#)
27. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, 3111–3119.
28. Alshahrani, M.; Khan, M.A.; Maddouri, O.; Kinjo, A.R.; Queralt-Rosinach, N.; Hoehndorf, R. Neuro-symbolic representation learning on biological knowledge graphs. *Bioinformatics* **2017**, *33*, 2723–2730. [\[CrossRef\]](#)
29. Gligorijević, V.; Barot, M.; Bonneau, R. deepNF: Deep network fusion for protein function prediction. *Bioinformatics* **2018**, *34*, 3873–3881. [\[CrossRef\]](#)
30. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S.Y. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4–24. [\[CrossRef\]](#)
31. Yan, K.; Lv, H.; Guo, Y.; Peng, W.; Liu, B. sAMPpred-GAT: Prediction of antimicrobial peptide by graph attention network and predicted peptide structure. *Bioinformatics* **2022**, *39*, btac715. [\[CrossRef\]](#) [\[PubMed\]](#)
32. You, R.; Yao, S.; Mamitsuka, H.; Zhu, S. DeepGraphGO: Graph neural network for large-scale, multispecies protein function prediction. *Bioinformatics* **2021**, *37*, i262–i271. [\[CrossRef\]](#)
33. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
34. Zhang, X.; Guo, H.; Zhang, F.; Wang, X.; Wu, K.; Qiu, S.; Liu, B.; Wang, Y.; Hu, Y.; Li, J. HNetGO: Protein function prediction via heterogeneous network transformer. *Briefings Bioinform.* **2023**, *24*, bbab556. [\[CrossRef\]](#)
35. Li, Q.; Han, Z.; Wu, X.M. Deeper insights into graph convolutional networks for semi-supervised learning. *Proc. AAAI Conf. Artif. Intell.* **2018**, *32*, 3438–3445. [\[CrossRef\]](#)
36. Chen, D.; Lin, Y.; Li, W.; Li, P.; Zhou, J.; Sun, X. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. *Proc. AAAI Conf. Artif. Intell.* **2020**, *34*, 3438–3445. [\[CrossRef\]](#)
37. Ying, C.; Cai, T.; Luo, S.; Zheng, S.; Ke, G.; He, D.; Shen, Y.; Liu, T.Y. Do transformers really perform badly for graph representation? *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 28877–28888.
38. Wu, Q.; Zhao, W.; Li, Z.; Wipf, D.; Yan, J. NodeFormer: A Scalable Graph Structure Learning Transformer for Node Classification. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, 28 November–9 December 2022.

39. Wu, Q.; Yang, C.; Zhao, W.; He, Y.; Wipf, D.; Yan, J. DIFFormer: Scalable (Graph) Transformers Induced by Energy Constrained Diffusion. In Proceedings of the International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023.
40. Wu, Q.; Zhao, W.; Yang, C.; Zhang, H.; Nie, F.; Jiang, H.; Bian, Y.; Yan, J. SGFormer: Simplifying and Empowering Transformers for Large-Graph Representations. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, 10–16 December 2023.
41. 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* **2023**, *39*, btad123. [\[CrossRef\]](#)
42. Lu, J.; Batra, D.; Parikh, D.; Lee, S. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In *Proceedings of the Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
43. Peng, Z.; Huang, W.; Gu, S.; Xie, L.; Wang, Y.; Jiao, J.; Ye, Q. Conformer: Local features coupling global representations for visual recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 367–376.
44. Jiao, P.; Wang, B.; Wang, X.; Liu, B.; Wang, Y.; Li, J. Struct2GO: Protein function prediction based on graph pooling algorithm and AlphaFold2 structure information. *Bioinformatics* **2023**, *39*, btad637. [\[CrossRef\]](#) [\[PubMed\]](#)
45. Wang, W.; Shuai, Y.; Li, Y.; Zeng, M.; Li, M. Enhancing Protein Function Prediction Through the Fusion of Multi-Type Biological Knowledge With Protein Language Model and Graph Neural Network. *IEEE Trans. Comput. Biol. Bioinform.* **2025**, *22*, 581–590. [\[CrossRef\]](#)
46. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
47. Gligorijević, V.; Renfrew, P.D.; Kosciolk, T.; Leman, J.K.; Berenberg, D.; Vatanen, T.; Chandler, C.; Taylor, B.C.; Fisk, I.M.; Vlamakis, H.; et al. Structure-based protein function prediction using graph convolutional networks. *Nat. Commun.* **2021**, *12*, 3168. [\[CrossRef\]](#) [\[PubMed\]](#)
48. You, R.; Zhang, Z.; Xiong, Y.; Sun, F.; Mamitsuka, H.; Zhu, S. GOLabeler: Improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics* **2018**, *34*, 2465–2473. [\[CrossRef\]](#)
49. You, R.; Yao, S.; Xiong, Y.; Huang, X.; Sun, F.; Mamitsuka, H.; Zhu, S. NetGO: Improving large-scale protein function prediction with massive network information. *Nucleic Acids Res.* **2019**, *47*, W379–W387. [\[CrossRef\]](#)
50. Verkuil, R.; Kabeli, O.; Du, Y.; Wicky, B.I.; Milles, L.F.; Dauparas, J.; Baker, D.; Ovchinnikov, S.; Sercu, T.; Rives, A. Language models generalize beyond natural proteins. *BioRxiv* **2022**. [\[CrossRef\]](#)
51. Szklarczyk, D.; Gable, A.L.; Nastou, K.C.; Lyon, D.; Kirsch, R.; Pyysalo, S.; Doncheva, N.T.; Legeay, M.; Fang, T.; Bork, P.; et al. The STRING database in 2021: Customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Res.* **2020**, *49*, D605–D612. [\[CrossRef\]](#)
52. Szklarczyk, D.; Gable, A.L.; Lyon, D.; Junge, A.; Wyder, S.; Huerta-Cepas, J.; Simonovic, M.; Doncheva, N.T.; Morris, J.H.; Bork, P.; et al. STRING v11: Protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Res.* **2019**, *47*, D607–D613. [\[CrossRef\]](#) [\[PubMed\]](#)
53. Huntley, R.P.; Sawford, T.; Mutowo-Meullenet, P.; Shypitsyna, A.; Bonilla, C.; Martin, M.J.; O'Donovan, C. The GOA database: Gene ontology annotation updates for 2015. *Nucleic Acids Res.* **2015**, *43*, D1057–D1063. [\[CrossRef\]](#) [\[PubMed\]](#)
54. Mitchell, A.L.; Attwood, T.K.; Babbitt, P.C.; Blum, M.; Bork, P.; Bridge, A.; Brown, S.D.; Chang, H.Y.; El-Gebali, S.; Fraser, M.I.; et al. InterPro in 2019: Improving coverage, classification and access to protein sequence annotations. *Nucleic Acids Res.* **2019**, *47*, D351–D360. [\[CrossRef\]](#) [\[PubMed\]](#)
55. Jones, P.; Binns, D.; Chang, H.Y.; Fraser, M.; Li, W.; McAnulla, C.; McWilliam, H.; Maslen, J.; Mitchell, A.; Nuka, G.; et al. InterProScan 5: Genome-scale protein function classification. *Bioinformatics* **2014**, *30*, 1236–1240. [\[CrossRef\]](#)
56. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
57. Hu, W.; Ohue, M. SpatialPPIv2: Enhancing protein–protein interaction prediction through graph neural networks with protein language models. *Comput. Struct. Biotechnol. J.* **2025**, *27*, 508–518. [\[CrossRef\]](#)
58. Holm, L.; Sander, C. Mapping the protein universe. *Science* **1996**, *273*, 595–602. [\[CrossRef\]](#)
59. Krissinel, E. On the relationship between sequence and structure similarities in proteomics. *Bioinformatics* **2007**, *23*, 717–723. [\[CrossRef\]](#)
60. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Židek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.