Integrating Heterogeneous Biological Networks and Ontologies for Improved Protein Function Prediction with Graph Neural Networks

Nhat C. Tran
Dept. of Computer Science and Engineering
The University of Texas at Arlington
Arlington, TX, USA
Email: nhat.tran@mavs.uta.edu

Jean X. Gao
Dept. of Computer Science and Engineering
The University of Texas at Arlington
Arlington, TX, USA
*Email: gao@uta.edu

***Abstract*—Elucidating protein functions is critical to advance our understanding of biological systems. However, the majority of proteins lack functional annotations due to the pace of manual knowledge-driven curation of the Gene Ontology (GO) terms. Automatic function prediction (AFP) aims to computationally infer these missing annotations by integrating sequence, interaction, and ontology data. Current AFP methods have yet to fully utilize heterogeneous relationships across multi-omics networks between non-coding RNAs, proteins, and the GO hierarchies. To address this, we introduce LATTE2GO, a novel graph neural network that integrates protein, RNA, and GO function entities and their interactions into a unified representation. By extracting higher-order associations with attention mechanism, LATTE2GO achieved significant gains over previous graph-based AFP techniques on CAFA4 benchmarks. Our analyses revealed modeling specific protein-protein interactions (PPI) and GO relationships increases accuracy in predicting molecular functions and biological processes. Overall, LATTE2GO demonstrates that heterogeneous graph neural networks can integrate diverse omics knowledge to advance systems-level understanding of protein roles within the complex milieu of functional and physical interaction networks.**

***Index Terms*—Heterogeneous graphs, multi-omics, Gene Ontology, protein-protein interactions, graph neural networks.**

I. Introduction

Proteins, being the building blocks of life, are central to nearly all molecular functions [1]. A comprehensive understanding of protein functions is crucial for advancing biological insights. Despite the tremendous growth in the number of identified protein sequences due to high-throughput technologies, functional annotations for the vast majority of proteins remain partly or entirely unknown. Therefore, AFP is a promising *in silico* approach to bridge this knowledge gap, especially where biochemistry experiments are constrained by time, cost, and expertise [2].

The AFP task is formalized through the Critical Assessment of Functional Annotation (CAFA) challenge [3], offering benchmarks for validating protein-function associations. Function terms are standardized by the Gene Ontology (GO) Consortium into three hierarchical ontologies: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC). The AFP prediction task is a challenging multi-label classification problem due to the large number of terms, varied annotation frequencies, and complex inter-term relationships. Additionally, there are multiple relationships among terms that exist within and between the MF, BP, and CC hierarchies, e.g., "is_a", "part_of", "has_part", "up_regulates", "down_regulates", etc., which together form a heterogeneous graph. Without considering these complexities, an AFP method can only extract data from a single GO relationship type or predict functions on a single ontology.

Learning protein representations from diverse data sources is another challenge in AFP. Unlike the classical view focusing solely on a single protein molecule's structure and function, protein interactomics reveals the complex network of interactions in which proteins operate [4] between multiple interaction networks. This holistic perspective emphasizes the contextual nature of protein functions within an extensive network of interactions, where a protein's function is the context of its physical interactions, genetic interactions, and other functional associations [5]. The effectiveness of this approach for AFP is signified in recent works, where integrated protein features from multi-modal data with PPI networks [6], [7] are more likely to outperform the approaches that rely on a single data type. Given the critical roles of other biomolecules like non-coding RNAs in biological networks, incorporating multi-omics interactions could further reveal associations vital for elucidating protein functions, as complex biological events usually involve the interplay of genes, transcripts, and proteins [8]. Even when there are no direct interactions between certain types of RNAs and proteins, it is possible to interrogate the indirect multi-hop relationships by exploring and identifying salient associations. Such networks, with nodes representing RNAs or proteins and edges indicating interactions or functional associations, present a flexible model for capturing the complexity of the underlying data.

To model the complexity of GO terms and multi-omics interactions, we propose aggregating these entities into a knowledge graph with heterogeneous relationships. Utilizing graph neural networks, our method, named LATTE2GO, ex-

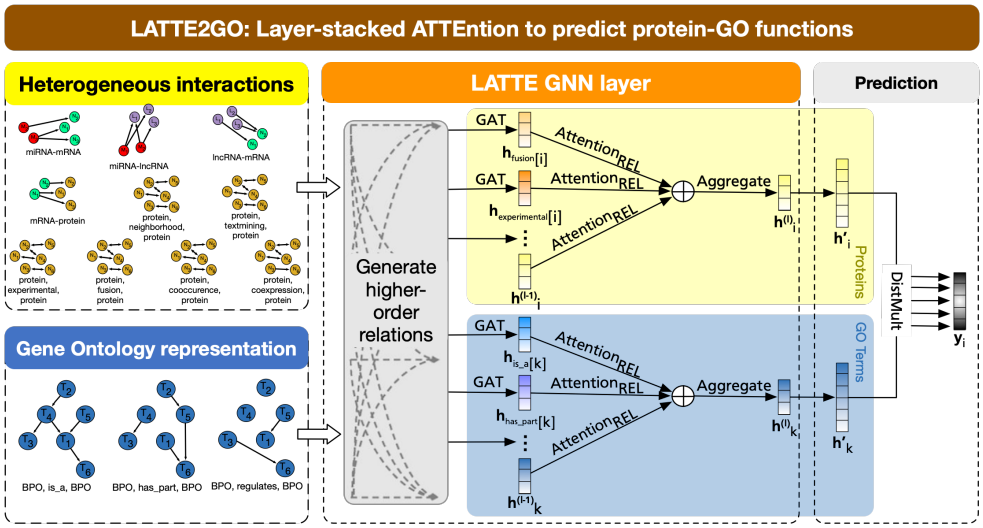

Fig. 1: The LATTE2GO architecture aggregates heterogeneous relations up to k-hops around a pair of protein and GO term in an integrated graph.

tracts valuable information from the graph structures among RNAs, proteins, and GO terms. It combines multiple data sources, including sequence features and interaction networks into a unified framework. More specifically, we built a network of Protein, MessengerRNA (mRNA), MicroRNA (miRNA), and Long non-coding RNA (LncRNA) heterogeneous interactions, along with GO terms. Key contributions of LATTE2GO include: (1) extracting higher-order multi-omics relationships from RNA-protein interactions and multi-relational protein-protein associations; (2) learning GO function representations from multiple hierarchical ontologies; and (3) leveraging attention graph neural networks for effective aggregation of heterogeneous PPI and GO term relationships. Through these mechanisms, LATTE2GO facilitates the inference of functional properties of proteins from complex, large-scale heterogeneous interaction networks, laying a robust foundation for enhanced protein function predictions.

II. Related Work

Several graph-based methods have tackled the AFP problem utilizing protein network data. For instance, You et al. [9] introduced DeepGraphGO, an end-to-end model employing two GCN layers to integrate sequence-based protein features and the STRING database's PPI network [10]. Similarly, DeepFunc [7] leveraged protein sequence data and DeepWalk to derive protein representations from a merged PPI network of STRING and BioGRID. These methods, however, overlook the differentiation between types of protein-protein associations by treating STRING PPI as homogeneous interactions, and by conflating physical and genetic interactions into a single graph. We posit that retaining the semantic information of specific interaction mechanisms through heterogeneous graph structures can extract richer information from protein networks. Although recent advancements in multi-relational and heterogeneous graph neural networks [11] offer promising avenues, they haven't been extensively explored in AFP literature nor have they demonstrated the ability to generate higher-order relations in multi-omics data.

On another front, various AFP methods aim to harness the hierarchical structures of GO terms for improved protein function prediction. DeepGOZero [12] leverages ontology-derived axiom constraints to learn GO term representations, enabling zero-shot predictions. HashGO [13] explored the underlying GO term structure to boost protein function prediction performance. Unlike these methods, our approach learns features directly from the complete hierarchical ontology and connects with protein network relations in an end-to-end fashion, potentially offering a more integrated understanding of protein functions.

III. Methods

A. Data Integration

1) Heterogeneous Networks Construction: We constructed an integrated graph from experimentally-validated public interaction databases by collecting multiple interaction networks among different RNA types and proteins. The databases employed, along with the criteria for relationship type integration, are listed below. All miRNA, lncRNA, and mRNA transcript names were harmonized across databases to standardized identifiers, and proteins were indexed by the UniProt protein ID. The integrative process encompassed:

- **microRNA-mRNA interactions:** from miRTarBase version 9.0 [14] database, which has a total of 414,828 directed interactions, and from TarBase [15], which includes 966,000 interactions.
- **microRNA-lncRNA interactions:** from DIANA-lncBase Experimental v3 [15], containing a total of 64,943 directed interactions, and from RNAInter [16], conatining 72,261 interactions.

- **lncRNA-protein interactions:** from RNAInter [16], which contain a total of 12,082,426 interactions.
- **mRNA-Protein relationships:** representing the one-to-many mapping between mRNAs and proteins with 227,972 directed relationships, using the "gene_name" attribute of UniProtKB/Swiss-prot annotation.
- **Protein-protein networks**: Extracted from STRING v11.5 database [10] with a subset of 239,987 proteins from 15 species, resulting in more than 45 million interactions.

We separated the different types of protein-protein associations into multiple sub-graphs where edges in the respective edge type have non-zero confidence score. Specifically, we obtained 17,556,841 "physical" interactions, which includes non-zero scores in either "experimental", "database", or "textmining" channels, 23,818,564 "co-expression" associations, 724,806 "co-occurrence" associations, 61,520 "fusion" associations, and 2,889,167 "neighborhood" associations to create five networks.

Utilizing the OpenOmics package [17], we efficiently combined all nodes and edges data into an integrated graph containing 86,927 lncRNAs, 199,025 mRNAs, 98,444 microRNAs, and 239,987 proteins. Overlapping edges were counted once when multiple networks were integrated for the same interaction type.

2) Gene Ontology Representation: We construct a heterogeneous graph to represent the entire Gene Ontology (GO) structure and the various relationships among GO terms, integrating it with the RNA-protein graph. The Gene Ontology data [18] was downloaded in OBO format, and the edge directionality was reversed for accurate representation. The selected relationship types include "is_a", "part_of", "has_part", "regulates", "positively_regulates", and "negative_regulates", where each encodes directed interactions, e.g., an edge $i \overset{is\_a}{\rightarrow} j$ signifies i as a parent term to j.

3) Protein Features: For each protein i is represented by a feature vector $\mathbf{x}_i$ generated from InterProScan, capturing the count of InterPro signature matches in the sequence. Specifically, $\mathbf{x}_i \in \mathbb{Z}^m$ is a sparse vector where m is the number of unique family, domain, and motif entries totaling 40,597 as of InterPro Release 90.0 [19]. A memory-efficient row-sparse matrix multiplication to obtain a low-dimensional vector representation $\mathbf{h}_i^{(0)} \in \mathbb{R}^d$ with $\mathbf{h}_i^{(0)} = \mathrm{ReLU}\left(\mathbf{W}^{(0)}\mathbf{x}_i + \mathbf{b}^{(0)}\right)$, where $\mathbf{W}^{(0)} \in \mathbb{R}^{d\times m}$ and $\mathbf{b}^{(0)} \in \mathbb{R}^d$ are learnable weights and biases, respectively. In our model, only the proteins' vector representations are extracted from feature attributes.

B. LATTE2GO GNN Architecture

The Layer-stacked ATTention Embedding to Gene Ontology (LATTE2GO) model architecture is illustrated in Figure 1.

1) Heterogeneous Graph Representation: We first define a heterogeneous directed graph $G = (\mathcal{V}, \mathcal{E}, \mathcal{T}, \mathcal{A})$ where each node $i \in \mathcal{V}$ and edge $e_{ij} \in \mathcal{E}$ are associated with their entity and relation type mapping function $\tau(i) : \mathcal{V} \to \mathcal{T}$ and $\phi(e_{ij}) : \mathcal{E} \to \mathcal{A}$, respectively. Generally $|\mathcal{T}| + |\mathcal{A}| \geq 2$, where $\mathcal{T}$ and $\mathcal{A}$ denote the sets of node and relation types.

a) Meta relations.: A directed edge e_{ij} links source node i to target node j with a meta relation denoted as $\langle \tau(i), \phi(e_{ij}), \tau(j)\rangle$. The set of all heterogeneous meta relation types is $\mathcal{A} = \{\langle s, r, t\rangle \mid s, t \in \mathcal{T}\}$, with the subset of edges with relation type r as $\mathcal{E}_r = \{e_{ij} \mid \phi(e_{ij}) = r\}$.

b) Higher-order meta relations.: We extend to higher-order meta-relations set $\mathcal{A}^{(l)}, l \geq 1$, representing l-hop meta-paths as sequences of l meta relations:

$$\mathcal{A}^{(l)} = \{\langle u, w \circ r, t\rangle \mid v = s, \langle u, w, v\rangle \in \mathcal{A}^{(l-1)}, \langle s, r, t\rangle \in \mathcal{A}\} \tag{1}$$

Here, $\mathcal{A}^{(1)} = \mathcal{A}$, and $\circ$ denotes the composition operator. The new edge set induced by a composed meta relation $r_c = r_a \circ r_b$ is $\mathcal{E}_{r_c} = \{e_{ik} \mid \phi(e_{ik}) = r_c, e_{ij} \in \mathcal{E}_{r_a}, e_{jk} \in \mathcal{E}_{r_b}\}$.

c) Knowledge graph preprocessing.: We preprocess the ground-truth heterogeneous graph, encompassing LncRNA, MicroRNA, MessengerRNA, protein, and GO term node types, by handling the undirected and directed edges from the referenced databases as respective meta relations. For undirected meta relations $r_u \in \mathcal{A}$, like $\langle \textit{Protein}, \textit{experimental}, \textit{Protein}\rangle$, we ensure $e_{ji} \in \mathcal{E}$, for every $e_{ij} \in \mathcal{E}$ where $\phi(e_{ij}) = \phi(e_{ji}) = r_u$. For directed meta relations $r_d \in \mathcal{A}$, a separate "reverse" relation r_d^{-1} is injected into $\mathcal{A}$ along with its reverse edges $\{e_{ji} \mid \forall e_{ij}$ where $\phi(e_{ij}) = r_d$ and $\phi(e_{ji}) = r_d^{-1}\}$ into $\mathcal{E}$, to ensure message propagation between all node types while maintaining the directed-ness semantics of the meta-relations.

2) Layer-stacked Attention on Meta-relations: Addressing the multiplicity of relations tied to proteins and GO terms, we hypothesize attention mechanisms as suitable for identifying salient relations for classifying protein-function relationships. Our model, LATTE2GO, employs the message-passing GNN framework to organize messages from relation-specific neighborhoods into contextualized embeddings. In the (l)-th LATTE layer, each node i's representation $\mathbf{h}_i^{(l)} \in \mathbb{R}^d$ updates by aggregating context embeddings from multiple relations as follows:

$$\begin{aligned} \mathbf{h}_i^{(l)} &= \underset{\forall r \in \mathcal{A}^{(l)}_{\tau(i)}}{\mathbf{Agg}} \left(\mathbf{Att}_{\mathrm{REL}}\left(r, i\right) \cdot \mathbf{h}_{\mathbf{r}_i}^{(l)}\right) \\ \mathcal{A}^{(l)}_{\tau(i)} &= \{\langle s, r, t\rangle \in \mathcal{A}^{(l)} \mid t = \tau(i)\} \cup \{\langle \tau(i)\rangle\} \end{aligned} \tag{2}$$

where $\mathbf{h}_{\mathbf{r}_i}^{(l)} \in \mathbb{R}^d$ represents node i's context embedding from relation r, and $\mathcal{A}^{(l)}_{\tau(i)}$ contains all l-hop meta relations with the target node type $\tau(i)$, including the "self" node type $\langle \tau(i)\rangle$ to represent the self-connection. Note that we're able to aggregate meta-relations from multiple source types to each target type, thus not constrained by predefined metapaths where $s = t$.

The self-attentional function $\mathbf{Att}_{\mathrm{REL}}$ adaptively infers the relation attention coefficients for each target node i, with:

$$\begin{aligned} \mathbf{Att}_{\mathrm{REL}}\left(r, i\right) &= \underset{\forall r \in \mathcal{A}^{(l)}_{\tau(i)}}{\mathrm{Softmax}}\left(\beta(r, l, i) + \mu_r\right) \\ \beta(r, l, i) &= \mathbf{b}_r^{(l)\top} f\left(\left[\mathbf{h}_{\mathbf{r}_i}^{(l)} \| \mathbf{h}_{\langle \tau(i)\rangle_i}^{(l)}\right]\right) \\ \mathbf{h}_{\langle \tau(i)\rangle_i}^{(l)} &= \mathbf{W}_{\tau(i)}^{(l)} \mathbf{h}_i^{(l-1)} \end{aligned} \tag{3}$$

where $\mathbf{h}_{\mathbf{r}i}^{(l)} \in \mathbb{R}^d$, the output from the Graph Attention Network (GAT) model [20], represents node i's context embedding given its neighbors from edges in relation r. Additionally, $\mathbf{b}_r^{(l)} \in \mathbb{R}^{2d}$ and μ_r are the trainable attention vector and bias scalar for relation r, respectively. $\mathbf{W}_{\tau(i)}^{(l)} \in \mathbb{R}^{d\times d}$ is the trainable weight matrix for node type $\tau(i)$, $\|$ denotes the concatenation, and f is the $\mathrm{LeakyReLU}_{\alpha=0.2}$ activation function.

The aggregation step combines the context embeddings with a weighted summation, employing H separate attention heads for stability [20]:

$$\underset{\forall r\in\mathcal{A}_{\tau(i)}^{(l)}}{\mathbf{Agg}}(\cdot) = \mathrm{LayerNorm}\left(\mathrm{ReLU}\left(\Big\|_{h=1}^{H}\sum_{\forall r\in\mathcal{A}_{\tau(i)}^{(l)}}(\cdot)\right)\right) \quad (4)$$

where if $H > 1$, then $\mathbf{h}_i^{(l)}$ and all parameters have hidden dimension size divided by H and are separate for each head.

Given that each layer l output node representations that contain context information aggregated only from l-hop meta relations, the final embedding for node i is obtained by stacking $\mathbf{h}_i^{(l)}$ from the outputs of L layers, as $\mathbf{h}'_i = \|_{l=1}^{L}\mathbf{h}_i^{(l)}$, where $\mathbf{h}'_i \in \mathbb{R}^{dL}$ to be used for end-to-end training with downstream tasks.

3) Computing Classification Scores Between Proteins and GO Terms: For predicting functions of protein i, a sampled subgraph is obtained from up-to-L-hops neighborhood expansions [21] starting from a "seed nodes" set including protein i and target classes set $\mathcal{V}_{GO}$. Despite the absence of relations between i and $\mathcal{V}_{GO}$, node representations for proteins and GO terms are computed simultaneously in a single feed-forward pass of the LATTE layers. We employ DistMult [22] instead of a final MLP layer to score the probability for class $k \in \mathcal{V}_{GO}$ as:

$$\hat{y}_{ik} = \sigma\left(\mathbf{h}_i'^{\top}\mathbf{M}_{\tau(k)}\mathbf{h}'_k\right) \quad (5)$$

where σ is the sigmoid function, and $\mathbf{M}_{\tau(k)} \in \mathbb{R}^{dL\times dL}$ is a trainable diagonal matrix for the ontology type $\tau(k)$.

For semi-supervised node classification learning, we use the binary cross-entropy loss function:

$$\mathcal{L}(\mathbf{\Theta}) = -\frac{1}{|\mathcal{V}_P||\mathcal{V}_{GO}|}\sum_{i\in\mathcal{V}_P}\sum_{k\in\mathcal{V}_{GO}} y_{ik}\log(\hat{y}_{ik}) + (1-y_{ik})\log(1-\hat{y}_{ik}) \quad (6)$$

where $\mathbf{\Theta}$ represents the set of learnable parameters, $\mathcal{V}_P$ is the subset of protein nodes with ground-truth labels, and $y_{ik} \in \{0,1\}$ is the true binary indicator for protein i and function k.

C. Model Training and Implementation Details

For efficient training of LATTE2GO on a graph with 6.6M nodes and 71M edges, we employ mini-batch SGD with subgraph sampling [21] and HGSampling [23], keeping the node budget per layer equal to the batch size. High-order meta-relations generation is CPU-parallelized using a dynamic programming approach, leveraging efficient sparse matrix multiplications. To manage the expanding size of the higher-order relations set $\mathcal{A}^{(l)}$ as seen in Eq. 1, we set rules: (1) meta relations of identical source and target node types can compose only if they share the same edge type, and (2) filter meta-relations in the last layer to have the target node type as the "seed nodes". Additionally, (3) during the composition of a high-order meta-relation $r_c = r_a \circ r_b$ within a sampled subgraph, we subsample the edges in r_a and in r_c to maintain approximately K neighbors for each target node. With these heuristics, LATTE2GO's worst-case time complexity is $\mathcal{O}\left(|\mathcal{A}|^L\left(|\mathcal{V}|K^L + |\mathcal{V}|Kd\right) + |\mathcal{V}|Ld^2\right)$.

IV. Experimental Results

A. Dataset Characteristics

We utilized the protein-GO annotation dataset from DeepGraphGO's benchmark [9], constructed per the CAFA4 outline. The dataset comprises GO annotations for 239,987 UniProtKB-SwissProt protein sequences [24], with designated training, validation, and testing sets based on time splits before Jan. 2018, Dec. 2018, and Jan. 2020, respectively. The 'IDA', 'IPI', 'EXP', 'IGI', 'IMP', 'IEP', 'IC', and 'TA' evidence-coded annotation set were collected from SwissProt1 [25] and UniProtGOA [26], with parent terms-propagated annotations added for all child term annotations and alias GO terms replaced with canonical names [27]. The sample size characteristics across all models are detailed in Table I.

B. Experimental Settings

For generating node classification outcomes on the CAFA4 benchmark dataset, methods were trained and validated independently on the Biological Process (BP), Molecular Function (MF), and Cellular Component (CC) ontology. All models were trained on identical training protein-function annotations, with early-stopping monitored on the validation set annotations, and evaluations conducted on the same test set annotations. Employing a consistent evaluation protocol as in [9] facilitates direct comparison with competing methods presented in the referenced article.

1) Baseline Methods: We compare LATTE2GO with various AFP methods categorized into homologous sequence transfer, sequence-only representation learning, and GNN-based methods on homogeneous and heterogeneous PPI networks. The *LR-InterPro* method computes GO term class scores using a linear transform followed by a sigmoid activation on protein feature vectors extracted from InterPro features. *BLAST-KNN* employs BLAST to identify homologous proteins for a query protein sequence, propagating GO term labels to the query protein based on similarity scores. Sequence-based models like *DeepGOCNN* and *DeepGOPlus* leverage 1D CNN encoders for protein sequence representation, with the latter combining CNN with sequence similarity-based predictions. In the category of GNN-based methods, *DeepGraphGO* incorporates InterPro features and STRING database's homogeneous PPI network using graph convolutional networks (GCN). *R-GCN* extends this by utilizing multi-relational PPI associations and considering edge weights for node classification. *HGT* is a state-of-the-art heterogeneous GNN model considering heterogeneous attention over each edge type without accounting for edge weight information or GO graph representation learning. Our proposed models, *LATTE2GO-1* and *LATTE2GO-2*, differ

TABLE I: Sample size characteristics of dataset splits

Ontology	Terms	Train proteins	Valid. proteins	Test proteins
MFO	6868	51549	490	426
BPO	21381	85104	1570	925
CCO	2832	76098	923	1224

in the order of meta relations considered, with the former considering only first-order while the latter incorporates both first and second-order meta relations, focusing solely on protein-protein and GO-GO relations.

Common hyper-parameters across all methods include an embedding dimension size of 512 and early stopping triggered if the validation AUPR metric does not improve after five epochs. For DeepGraphGO, R-GCN, and HGT, a two-layer GNN is employed followed by an MLP layer for outputting node labels, with training conducted end-to-end on InterPro protein features. Regarding mini-batch subgraph sampling, while DeepGraphGO employs full-neighborhood expansion at each layer on a k-NN PPI subgraph with $k = 30$, R-GCN, HGT, and LATTE2GO utilize HGSampling [23] on the entire interaction set.

2) Evaluation Metrics: We employed F_{max} and AUPR (Area under Precision-Recall curve) metrics, as primary evaluation measures as per [3]. AUPR is pair-centric, evaluated over predicted scores of protein-function pairs across 100 thresholds between 0.0 and 1.0. F_{max}, a protein-centric measure, computes the maximum F1 score at any threshold on classification scores among all GO term classes, averaged over all proteins, defined as: $F_{max} = max_t \left\{ \frac{2 \cdot pr(t) \cdot rc(t)}{pr(t)+rc(t)} \right\}$, where $pr(t)$ and $rc(t)$ denote precision and recall at a positive-class threshold value t, respectively, as outlined in [9].

C. Comparison Results

Table II presents the performance comparison of LATTE2GO with baseline methods using F_{max} and AUPR metrics. In replicating the same experimental settings as in DeepGraphGO's article [9], LATTE2GO-1, R-GCN, and HGT exhibit substantial performance improvement in BPO and MFO over DeepGraphGO on both F_{max} and AUPR metrics. These methods also use InterPro protein features, but can leverage STRING's multi-relational protein-protein networks for enhanced protein representations. This observation supports our hypothesis that distinguishing between physical and genetic protein-protein associations coupled with specified genetic interaction types, enhances protein representation learning and possibly better unveils functional associations in varied biological contexts. Comparing LATTE2GO-1 with R-GCN and HGT shows a performance boost in BPO F_{max}. This suggests that the integration of GO's multi-relational associations graph could lead to improved GO term representations, maintaining competitive classification performance as in typical node classification settings, especially given the large size of the BPO target classes. Moreover, representing target classes as a graph posits a robust framework for inferring annotations on sparse or unannotated terms in future investigations.

TABLE II: Performance comparison results under DeepGraphGO's experimental settings.

Method	Fmax			AUPR		
	MFO	BPO	CCO	MFO	BPO	CCO
LR-InterPro	0.617	0.278	0.661	0.530	0.144	0.672
BLAST-KNN	0.590	0.274	0.650	0.455	0.113	0.570
DeepGOCNN	0.434	0.248	0.632	0.306	0.101	0.573
DeepGOPlus	0.593	0.290	0.672	0.398	0.108	0.595
DeepGraphGO	0.623	0.327	0.692	0.543	0.194	0.695
R-GCN	0.781	0.526	0.710	**0.836**	0.545	0.712
HGT	0.762	0.511	**0.716**	0.836	0.529	**0.736**
LATTE2GO-1	0.778	0.539	0.691	0.753	0.534	0.689
LATTE2GO-2	**0.840**	**0.574**	0.683	0.831	**0.584**	0.682

V. Discussion

The performance improvement in LATTE2GO-2 over LATTE2GO-1 highlights the efficacy of generating higher-order meta relations. Unlike multi-layer RGCN and HGT, which operate on a first-order graph structure across all layers, our approach dynamically generates and aggregates meta relations. This methodology offers dual advantages: (1) it decouples the higher-order metapath to retain semantic information in higher-order neighborhoods, and (2) mitigates the "over-squashing" issue [28] by preventing the merging of higher-order context with lower-order context across layers. Our experiments indicate that employing second-order relations yields satisfactory performance, aligning with the notion that proteins with two-hop neighbors in the interaction topology share similar characteristics and functions [29].

We additionally conducted an ablation study to evaluate the impact of various core components in LATTE2GO: node and interaction types selection, higher-order relationship generation, and concatenation of multiple higher-order embeddings. The study utilized a grid search executed via Weight and Biases to explore all setting combinations of these components, focusing on the BPO AUPR metrics on a test dataset of human- and mouse-only proteins.

As depicted in Fig. 2, the "Heterogeneous node types" plot shows a dip in performance when including RNA node types, with a more substantial decline when all seven node types are included. Conversely, the protein-only or protein-and-BPO configurations achieved the highest performance. This outcome suggests that multi-omics RNA interactions with proteins may not enhance function prediction in GNN-based models, yet underscores the efficacy of learning both proteins and GO representations within a single GNN architecture. In the "Split PPI interaction" types plot, we identified a notable BPO AUPR improvement when treating the STRING data as heterogeneous PPI, supporting our hypothesis of multi-relational PPI for more accurate AFP. Other findings indicate that generating second-order meta relations enhances AUPR, though the effect of concatenating layer embeddings in *LATTE2GO-2* remains ambiguous.

Fig. 2: Ablation analysis reporting differences on AUPR metric on (1^{st}) the heterogeneous graph node types, (2^{nd})separating STRING protein-protein associations, (3^{th}) generating higher-order meta-relations, and (4^{th}) concatenating layer embeddings.

VI. Conclusion

This paper introduced LATTE2GO, a heterogeneous graph neural network framework for automatic protein function prediction, leveraging an integrated graph of RNAs, proteins, and GO functions. Our method exploits the expressive representation of heterogeneous protein-protein associations alongside a GO knowledge graph, enabling a rich semantic context for deriving new relationships without the need for manually crafted features. The versatility of this graph-based approach opens avenues for enhanced AFP, especially when integrating more complex data structures like the Enzyme Commission Ontology and InterPro entities. Furthermore, the incorporation of additional relations can enrich annotations with context-specific information regarding the cellular context of protein functions. The promising results from LATTE2GO suggest potential for further exploration, especially in inductive prediction settings with sparse protein interactions or missing annotations on specific GO terms, paving the path for more comprehensive and accurate protein function predictions.

References

[1] D. S. Goodsell, "The machinery of life," 2009.

[2] R. Ramola, I. Friedberg, and P. Radivojac, "The field of protein function prediction as viewed by different domain scientists," *bioRxiv*, 2022.

[3] N. Zhou, Y. Jiang, T. R. Bergquist, A. J. Lee, B. Z. Kacsoh, A. W. Crocker, K. A. Lewis, G. Georghiou, H. N. Nguyen, M. N. Hamid *et al.*, "The cafa challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens," *Genome biology*, vol. 20, no. 1, pp. 1–23, 2019.

[4] R. Bonetta and G. Valentino, "Machine learning techniques for protein function prediction," *Proteins: Structure, Function, and Bioinformatics*, vol. 88, no. 3, pp. 397–413, 2020.

[5] D. Guala, C. Ogris, N. Müller, and E. L. Sonnhammer, "Genome-wide functional association networks: background, data & state-of-the-art resources," *Briefings in bioinformatics*, vol. 21, no. 4, pp. 1224–1237, 2020.

[6] M. N. Wass, G. Barton, and M. J. Sternberg, "Combfunc: predicting protein function using heterogeneous data sources," *Nucleic acids research*, vol. 40, no. W1, pp. W466–W470, 2012.

[7] F. Zhang, H. Song, M. Zeng, Y. Li, L. Kurgan, and M. Li, "Deepfunc: a deep learning framework for accurate prediction of protein functions from protein sequences and interactions," *Proteomics*, vol. 19, no. 12, p. 1900019, 2019.

[8] C. Monti, M. Zilocchi, I. Colugnat, and T. Alberio, "Proteomics turns functional," *Journal of proteomics*, vol. 198, pp. 36–44, 2019.

[9] R. You, S. Yao, H. Mamitsuka, and S. Zhu, "Deepgraphgo: graph neural network for large-scale, multispecies protein function prediction," *Bioinformatics*, vol. 37, no. Supplement_1, pp. i262–i271, 2021.

[10] D. Szklarczyk, A. L. Gable, K. C. Nastou, D. Lyon, R. Kirsch, S. Pyysalo, N. T. Doncheva, M. Legeay, T. Fang, P. Bork *et al.*, "The string database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets," *Nucleic acids research*, vol. 49, no. D1, pp. D605–D612, 2021.

[11] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The world wide web conference*, 2019, pp. 2022–2032.

[12] M. Kulmanov and R. Hoehndorf, "DeepGOZero: improving protein function prediction from sequence and zero-shot learning based on ontology axioms," *Bioinformatics*, vol. 38, pp. i238–i245, 06 2022.

[13] G. Yu, Y. Zhao, C. Lu, and J. Wang, "Hashgo: hashing gene ontology for protein function prediction," *Computational biology and chemistry*, vol. 71, pp. 264–273, 2017.

[14] H.-Y. Huang, Y.-C.-D. Lin, S. Cui, Y. Huang, Y. Tang, J. Xu, J. Bao, Y. Li, J. Wen, H. Zuo *et al.*, "mirtarbase update 2022: an informative resource for experimentally validated mirna–target interactions," *Nucleic acids research*, vol. 50, no. D1, pp. D222–D230, 2022.

[15] D. Karagkouni, M. D. Paraskevopoulou, S. Chatzopoulos, I. S. Vlachos, S. Tastsoglou, I. Kanellos, D. Papadimitriou, I. Kavakiotis, S. Maniou, G. Skoufos *et al.*, "Diana-tarbase v8: a decade-long collection of experimentally supported mirna–gene interactions," *Nucleic acids research*, vol. 46, no. D1, pp. D239–D245, 2018.

[16] J. Kang, Q. Tang, J. He, L. Li, N. Yang, S. Yu, M. Wang, Y. Zhang, J. Lin, T. Cui *et al.*, "Rnainter v4. 0: Rna interactome repository with redefined confidence scoring system and improved accessibility," *Nucleic acids research*, vol. 50, no. D1, pp. D326–D332, 2022.

[17] N. C. Tran and J. X. Gao, "Openomics: A bioinformatics api to integrate multi-omics datasets and interface with public databases." *Journal of Open Source Software*, vol. 6, no. 61, p. 3249, 2021.

[18] T. G. O. Consortium, "The gene ontology resource: enriching a gold mine," *Nucleic acids research*, vol. 49, no. D1, pp. D325–D334, 2021.

[19] T. Paysan-Lafosse, M. Blum, S. Chuguransky, T. Grego, B. L. Pinto, G. A. Salazar, M. L. Bileschi, P. Bork, A. Bridge, L. Colwell *et al.*, "Interpro in 2022." *Nucleic Acids Research*, 2022.

[20] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[21] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[22] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," *arXiv preprint arXiv:1412.6575*, 2014.

[23] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proceedings of The Web Conference 2020*, 2020, pp. 2704–2710.

[24] U. Consortium, "Uniprot: a worldwide hub of protein knowledge," *Nucleic acids research*, vol. 47, no. D1, pp. D506–D515, 2019.

[25] E. Boutet, D. Lieberherr, M. Tognolli, M. Schneider, and A. Bairoch, "Uniprotkb/swiss-prot," in *Plant bioinformatics*. Springer, 2007, pp. 89–112.

[26] R. P. Huntley, T. Sawford, P. Mutowo-Meullenet, A. Shypitsyna, C. Bonilla, M. J. Martin, and C. O'Donovan, "The goa database: gene ontology annotation updates for 2015," *Nucleic acids research*, vol. 43, no. D1, pp. D1057–D1063, 2015.

[27] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[28] U. Alon and E. Yahav, "On the bottleneck of graph neural networks and its practical implications," *arXiv preprint arXiv:2006.05205*, 2020.

[29] H. N. Chua, W.-K. Sung, and L. Wong, "Exploiting indirect neighbours and topological weight to predict protein function from protein–protein interactions," *Bioinformatics*, vol. 22, no. 13, pp. 1623–1630, 2006.