, , pp. 1–10

GATSBI: Improving context-aware protein embeddings through biologically motivated data splits

Gowri Nayar [1] and Russ B. Altman [1, 2,3,4,*]

[1]Department of Biomedical Data Science, Stanford University, CA, USA
[2]Department of Bioengineering, Stanford University, CA, USA
[3]Department of Genetics, Stanford University, CA, USA
[4]Department of Medicine, Stanford University, CA, USA

*Corresponding author. Russ.Altman@stanford.edu

Abstract

Motivation - Understanding protein function requires integrating diverse biological evidence while accounting for strong contextual dependence. Recent protein embedding methods increasingly leverage heterogeneous biological networks, yet their evaluation protocols often fail to reflect the specific biological tasks for which the embeddings are intended. Prediction of missing interactions, annotation of new proteins, and discovery of functional modules require fundamentally different data partitions, such as edge-masked versus node-held-out splits. Moreover, most approaches report performance primarily on well-studied proteins, where computational predictions are least needed, risking substantial overestimation of real-world utility.

Results - We introduce a graph attention–based framework (Gatsbi) to construct context-aware protein embeddings from integrated protein–protein interactions, co-expression, sequence representations, and tissue-specific associations. Using task-aligned evaluation protocols, we show that models trained with biologically appropriate partitions achieve markedly better generalization. Across interaction, function, and functional set prediction, Gatsbi consistently outperforms existing pretrained embeddings for both well-studied and understudied proteins, with the largest gains observed for the understudied regime and under inductive node-held-out evaluation. To enable broad reuse, we provide the learned embeddings for download for application to other protein prediction tasks.

Availability and Implementation - Code and models for our experiments are available at https://github.com/Helix-Research-Lab/GATSBI-embedding

Keywords Heterogeneous protein networks, Graph attention networks, Context-aware protein embeddings

1. Introduction

Protein function depends on cellular and biological context [1]. To capture this complexity, computational models integrate heterogeneous data—such as protein–protein interactions, co-expression, transcriptomics, sequence and structural representations, and tissue-specific associations—to learn context-aware protein embeddings [2]. These embeddings place functionally related proteins nearby in a shared representation space, enabling downstream tasks including interaction prediction, functional annotation, and disease association. This is particularly critical for understudied proteins, which lack direct experimental characterization and must rely on information transfer from related, well-studied proteins [3]. Because most future predictions will concern such proteins, performance in this regime is the key measure of real-world utility. Effective embeddings must therefore fuse diverse modalities while preserving the relational structure of biological data, and their growing use in translational applications demands rigorous, context-aware evaluation to ensure reliability [4].

Regarding protein function, the biological evidence is inherently relational, consisting of pairwise relationships such as physical interactions and co-expression patterns. Therefore, a natural method to model this data is as a biological network, where proteins are represented as nodes, and the interactions are edges. A heterogeneous network can integrate multiple data sources together[5]; the edges are labeled by the type of biological relationship, such as physical interactions, co-expression, and tissue-specific associations. This representation preserves both the diversity of biological evidence and the structure of functional relationships between proteins[6]. Learning effective protein representations from such networks requires models that can propagate and integrate information across different edge types and biological contexts.

Graph attention networks (GATs) can be used to generate an embedding vector from this heterogeneous network[7]. A GAT is a type of graph neural network that learns a vector representation for each protein by repeatedly updating its features based on information from neighboring proteins in the network. At each layer of the model, a protein computes a new representation by collecting information

© The Author(s) . Copyright and licence statements to be updated by the publisher during production.

from its directly connected neighbors, called message passing. Rather than treating all neighbors equally, the model assigns an attention weight to each neighbor, which indicates how important that neighbor's information is for updating the protein's representation. These attention weights are learned from the data and depend on both the interacting proteins and the type of biological relationship connecting them. The model then uses this attention value to perform a weighted combination of the neighboring information and applies a nonlinear transformation to produce the updated embedding. By stacking multiple layers, the model allows information to propagate across the network, so each protein's final embedding captures both its local interactions and its broader biological context.

Pinnacle is currently the only method that combines heterogeneous biological context, including tissue- and cell-specific information, and that provides publicly available pretrained embeddings that can be used directly for downstream prediction[8]. By incorporating contextual signals beyond sequence and interaction topology, Pinnacle produces representations that better reflect the condition-dependent nature of protein function, and that can be readily transferred to a wide range of predictive tasks. The availability of these pretrained embeddings has substantially increased the practical impact of representation learning in biology, enabling researchers to perform tasks such as interaction prediction, functional annotation, and disease association without the computational burden of training a new embedding model for each application. However, Pinnacle is trained and evaluated using random data splits, which limits robustness to diverse biological tasks[9].

The way biological information is partitioned during training and evaluation of such embedding models often fails to reflect the true settings in which biological predictions are ultimately made[10]. Because embeddings are learned directly from the available biological evidence, the choice of data partitioning fundamentally determines what information the model is allowed to observe and therefore what type of generalization it can achieve[11]. In practice, reliance on a single random data partition is often insufficient to produce embeddings that generalize reliably to unseen proteins or novel biological contexts, potentially leading to overoptimistic estimates of downstream performance[12].

An underexplored issue is the disparity between well-studied proteins, which have direct experimental evidence and curated annotations, and understudied proteins, for which little experimental information is available[13]. Current biological networks and benchmarks are biased toward well-studied proteins, since these are the proteins for which interaction data and functional annotations are most readily available[14]. As a result, models trained and assessed primarily on this subset may not reflect performance in settings where many proteins remain poorly characterized. Analyzing model performance separately on well-studied and understudied proteins can better show how learned embeddings support inference in data-limited regimes[14].

In this work, we introduce a protein embedding framework, Gatsbi (GAT with Split Boosted Inference), that represents heterogeneous biological data as a network and learns context-aware embeddings using graph attention–based message passing. We first build a heterogeneous network that integrates protein-protein interactions, co-expression relationship, tissue-specific association, and sequence derived features. We use this network to train a graph attention network using two different data splitting strategies, edge split and node split. We then evaluate the embeddings using different prediction tasks - interaction prediction, protein function prediction, and functional group prediction. We also evaluate the impact of performance in both well-studied and understudied proteins. We find that the performance across the tasks are dependent on the data split used. Interaction prediction has a higher accuracy with the edge split embedding (0.79) than with the node split embedding (0.64); the opposite is true for the node function prediction task. Furthermore, for both the Gatsbi and the baseline embeddings, the performance is statistically significantly higher for well-studied (high degree) proteins compared to understudied (low degree) proteins, but the biologically meaningful data split creates a smaller performance difference than a random split. This work therefore addresses both the construction of protein embeddings and the experimental conditions under which their performance can be reliably assessed.

Key Takeaways:

1. This work demonstrates that embedding construction and data partitioning influence task-specific generalization. By controlling the biological information available during embedding learning, we show how different partitioning strategies affect performance across distinct prediction tasks.
2. We characterize differences in generalization between well-studied and understudied proteins. Separate analysis of these regimes highlights how learned embeddings behave in data-limited settings.
3. We provide the generated protein embeddings that can be used for downstream prediction tasks.

2. Methods

2.1 Heterogeneous data sources capture complementary information across distinct functional aspects of protein behavior.

2.1.1 Protein sequence representations

We encode protein sequences using the Evolutionary Scale Modeling (ESM-2) protein language model, a large-scale transformer trained on millions of protein sequences to learn context-aware representations that capture evolutionary, structural, and functional properties of proteins[15]. For each protein, the corresponding amino acid sequence is passed through the pretrained ESM-2 model, and the resulting per-residue embeddings from the final hidden layer are aggregated using mean pooling to obtain a fixed-length protein-level representation. Thus, we obtain a vector representation for 20,074 human proteins.

2.1.2 Human protein interaction network

The reference protein-protein interaction (PPI) network is the STRING network[16]. From the STRING database, we first obtain all human protein interactions and the associated score for each interaction. We use this data to obtain data on physical protein interactions, so we filter all STRING relationship to those that have experimental evidence of physical interaction. Reference PPI databases are known to have noisy interactions[17], and so we filter out interactions with an associated score of less than 0.6[18]. This results in 16,462 number of protein nodes and 217,092 interactions.

2.1.3 Protein co-expression patterns

Coordinated expression across biological contexts reflects shared regulatory control and pathway involvement, making co-expression a widely used indicator of functional coupling between proteins[19]. We obtain protein co-expression relationships from the STRING

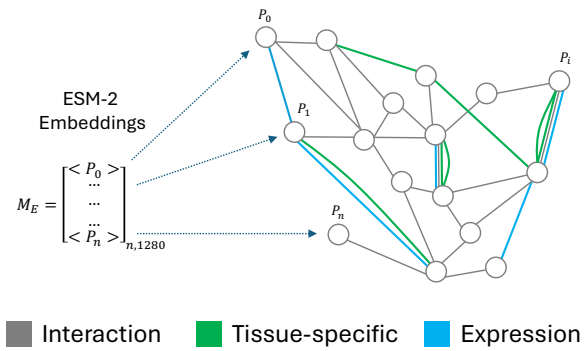

Figure 1 We constructed a heterogeneous protein network integrating sequence representations, physical interactions, co-expression, and tissue-specific associations. Nodes P_0–P_n represent proteins in the human proteome and are initialized with ESM-2 embeddings $M_E \in \mathbb{R}^{n \times 1280}$, providing one sequence vector per protein. Edges encode functional associations: gray edges denote physical interactions (STRING experimental channel), blue edges denote co-expression relationships (STRING co-expression channel), and green edges denote tissue-specific associations (HumanBase). Each edge is annotated with its data source, association score, and tissue type when applicable, and multiple labeled edges may connect the same protein pair.

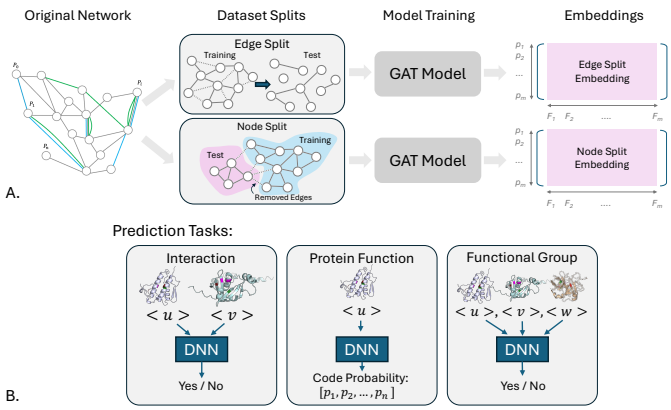

Figure 2 Overview of the evaluation framework. The original heterogeneous protein network is partitioned using either an edge split or a node split, producing distinct training and test graphs. A GAT model is trained on each split to generate protein embeddings, which are then used in downstream prediction tasks including interaction prediction, protein function classification, and functional group prediction.

database using its co-expression evidence channel, which integrates large-scale transcriptomic and proteomic data across diverse biological contexts and converts these signals into calibrated functional association scores[16]. Again, we filter out relationships that have an association score of less than 0.6. This results in 11,205 proteins with 151,067 pairwise co-expression relationships.

2.1.4 Tissue-specific functional associations

We additionally use HumanBase as a source of context-specific functional interaction networks[20]. HumanBase integrates bulk transcriptomic data with regulatory and interaction annotations to infer probabilistic gene–gene functional associations across 144 human tissue and cell-type contexts. We download the precomputed networks, retain high-confidence associations (posterior probability ≥ 0.6), and map gene identifiers to UniProt protein identifiers to obtain 17,617 proteins with 1,207,151 tissue-specific, functional relationships.

2.2 Integrating data sources into heterogeneous network

We model the complete set of pairwise protein relationships (interactions, expression, tissue-specific association) as a network, where the nodes correspond to proteins and an edge is present between any two nodes that have a pairwise relationship. We initialize the protein nodes with the sequence representation vector obtained from ESM-2. The edges are labeled with the data source from which the pairwise relationship originates(interaction, co-expression, tissue-specific association). If the edge originates from HumanBase, and thus is a tissue-specific relationship, the tissue type is also saved on the edge. Each edge is also annotated with the association score. The graph allows for multiple edges between the same pair of nodes, so a pair of proteins u and v can have multiple edges with different attributes (data source, weight, tissue type) connecting them. The final network has 18,049 nodes and 1,575,310 edges. Figure 1 displays a diagram describing the intergration of the data sources into one heterogeneous network.

2.3 Graph Attention Network

The constructed heterogeneous protein network is used as input to the GATSBI graph attention network (GAT), which learns a low-dimensional embedding for each protein node by integrating the network information. The heterogeneous network is denoted as $G = (V, E, \mathcal{T})$, where V is the set of protein nodes, E is the set of weighted edges, and $\mathcal{T}$ is the set of edge types corresponding to physical interactions, co-expression, and tissue-specific functional associations. Each node $v \in V$ is initialized with its ESM-2 sequence embedding $\mathbf{h}_v^{(0)} \in \mathbb{R}^d$.

At each GAT layer ℓ, node representations are updated via attention-based message passing over heterogeneous edge types. For a node v, the updated representation $\mathbf{h}_v^{(\ell+1)}$ is computed as

$$\mathbf{h}_v^{(\ell+1)} = \sigma\left(\sum_{u \in \mathcal{N}(v)} \alpha_{vu}^{(\ell)} \mathbf{W}_{\tau(v,u)}^{(\ell)} \mathbf{h}_u^{(\ell)}\right), \tag{1}$$

where $\tau(v,u) \in \mathcal{T}$ denotes the edge type between v and u, $\mathbf{W}_\tau^{(\ell)}$ is a type-specific transformation matrix, and $\alpha_{vu}^{(\ell)}$ is the attention coefficient for message propagation along that edge.

The attention coefficient is factorized as

$$\alpha_{vu}^{(\ell)} \propto \underbrace{\mathrm{Attn}^{(\ell)}\left(\mathbf{h}_v^{(\ell)}, \mathbf{h}_u^{(\ell)}, \mathbf{e}_{vu}\right)}_{\text{learned compatibility}} \cdot \underbrace{p_{\tau(v,u)}^{(\ell)}}_{\text{edge-type prior}} \cdot \underbrace{q_{c(v),c(u)}^{(\ell)}}_{\text{tissue-consistency prior}}, \tag{2}$$

followed by normalization over the neighborhood of v.

Here, $p_\tau^{(\ell)}$ is a learnable scalar prior that controls the overall probability of traversing each edge type. The term $q_{c(v),c(u)}^{(\ell)}$ biases propagation toward edges whose tissue context matches the currently active tissue context $c(v)$ of node v. When propagation follows a tissue-specific edge of type t, edges belonging to the same tissue context t are up-weighted relative to edges of other tissue contexts in subsequent message-passing steps, enforcing tissue-consistent information flow.

After L layers of propagation, the final embedding $\mathbf{z}_v = \mathbf{h}_v^{(L)}$ represents the learned protein embedding,.

2.4 Training details for GATSBI

GATSBI learns protein representations from the heterogeneous network by optimizing a self-supervised link prediction objective that preserves relationships across data sources. During training, a subset of edges is masked, and the model is trained to predict their presence and type while discriminating against negative samples, using graphs that exclude the masked edges for training, validation, and testing. In the following subsections, we describe the construction of task-specific data splits and the downstream tasks used to evaluate the learned embeddings. Figure 2 A shows the model architecture overview, including the data split and the model.

2.4.1 Task-driven graph splits for interaction-level and protein-level generalization

To rigorously evaluate generalization under different biological scenarios, we construct complementary data-splitting strategies - (1) edge (relationship-level) split and (2) node (protein-level) split.

In the edge split, we assign 70% of protein pairs to training and 30% to testing while retaining the full set of protein nodes and approximately preserving the per-datasource edge proportions. To minimize topological leakage, we construct the split such that, after removing all test edges, the shortest path between any test pair is at least 10 hops in the training graph. Because the network is a multigraph, we also group edges by protein pair: all relation types between a held-out pair are removed from training and placed exclusively in the test set.

In the node split (inductive), we partition proteins into 70% training nodes and 30% test nodes and construct a training graph containing only V_{train} and the edges between them, removing all edges incident to test nodes so that no direct interaction evidence about these proteins is available during representation learning. The link-prediction objective is optimized exclusively on this subgraph: a subset of edges among training nodes is masked, and the model is trained to recover these edges while discriminating against negative pairs sampled from V_{train}, ensuring that no parameters are influenced by test-node information. To further prevent information transfer through sequence homology, we enforce less than 30% sequence similarity between proteins in the two sets, so that test proteins cannot be trivially inferred from closely related training proteins. At inference, the frozen encoder is applied inductively to compute embeddings for test proteins by inserting them with their observed features and performing a forward pass using only parameters learned from V_{train}; link prediction involving test nodes is then performed without additional training. This setting therefore evaluates generalization to genuinely understudied proteins that lacked both interaction and sequence-based evidence during model fitting, reflecting real-world scenarios in which new or poorly characterized proteins must be interpreted using contextual structure learned from other parts of the network.

To unify the data-splitting strategies under a single training framework, we explicitly distinguish between: (1) the set of nodes available for message passing, V_{mp}, (2) the set of edges available for message passing, E_{mp}, and (3) the set of positive edges scored by the training loss, E^{+}_{loss}.

For the edge split, all nodes participate in message passing, $V_{\mathrm{mp}} = V$, while the message-passing graph contains only training edges, $E_{\mathrm{mp}} = E_{\mathrm{train}}$; the same edges define the positive loss set, $E^{+}_{\mathrm{loss}} = E_{\mathrm{train}}$. This configuration measures recovery of held-out relationships among known proteins.

For the node split (inductive), message passing is restricted to training proteins, $V_{\mathrm{mp}} = V_{\mathrm{train}}$, with edges limited to those induced by these nodes, $E_{\mathrm{mp}} = E[V_{\mathrm{train}}]$; the loss is likewise computed only on this subgraph, $E^{+}_{\mathrm{loss}} = E[V_{\mathrm{train}}]$. This configuration measures generalization to proteins absent during training.

For each training batch, negative edges are generated using a node-degree–matched sampling strategy rather than uniform random pairing. Specifically, for each positive edge in E^{+}_{loss}, we sample non-adjacent protein pairs whose endpoint degrees closely match those of the corresponding positive pair, thereby controlling for degree-dependent biases in the link-prediction objective. Candidate negatives are drawn from V_{mp} while ensuring that they do not overlap with any known positive edges from the full network. We sample five such negatives per positive edge, maintaining a fixed negative-to-positive ratio of 5:1 across all experiments.

2.4.2 Creating the protein embeddings

Using the data splits and optimization described above, we train the GATSBI embedding model under a shared GAT-based link-prediction objective. For each split, the message-passing graph is constructed from the corresponding training portion of the network, using training edges for the edge split and the subgraph induced by V_{train} for the node split. Node features are initialized from 1280-dimensional pretrained ESM vectors, centered by subtracting the dataset centroid, and treated as trainable embeddings. These features are projected to 512 dimensions and passed through a three-layer GAT encoder with hidden size 512, attention heads $(4, 4, 2)$, and dropout 0.2 to produce 512-dimensional protein embeddings. Edge scores are computed using an MLP decoder with hidden size 128 and dropout 0.2, and the model is trained with a binary cross-entropy objective. Optimization uses the Adam optimizer with learning rate 10^{-3}, weight decay 10^{-4}, a cosine-annealing schedule with $T_{\max} = 50$, and gradient clipping at 2.0 for a maximum of 50 epochs. The resulting embeddings are saved after training and evaluated on the downstream prediction tasks described in Section 2.5.

2.5 Evaluating the protein embeddings

We evaluate the embeddings on three protein prediction tasks. Figure 2 B illustrates these three tasks.

2.5.1 Prediction tasks used to measure the utility of protein embeddings

We evaluate the utility of the learned protein embeddings across three complementary prediction tasks. We run each experiment 10 times with random seeds. In all cases, we deliberately use lightweight predictive models so that performance reflects the quality of the embeddings rather than the capacity of the downstream classifier.

Interaction prediction (edge prediction). To evaluate whether the embeddings encode information about physical and functional relationships, we perform protein–protein interaction prediction using BioGRID as the source of positive edges[21]. We filter all BioGRID edges to obtain a distinct set of edges from those included in the embedding creation. Each positive example corresponds to a curated physical interaction between two proteins. Negative examples are generated by randomly sampling protein pairs that do not appear in BioGRID. For each pair, we concatenate the two embedding vectors and train a shallow neural network classifier to predict whether an interaction exists. The task is evaluated under

an edge-level split to ensure that test interactions involve protein pairs not seen during training.

Protein function prediction (node prediction). We formulate enzyme function prediction as a multilabel node classification task using Enzyme Commission (EC) annotations obtained from UniProt[22]. Proteins with at least one EC label and an available embedding are retained. Each protein is represented by its learned embedding, and the task is evaluated under a node-level split to ensure that test proteins are unseen during training. A three-layer multilayer perceptron (MLP) is trained to predict the EC labels from the embedding vectors. This task assesses whether the embeddings capture functional biochemical properties of individual proteins.

Functional set prediction (pathway-level set classification). To assess whether the embeddings capture higher-order biological organization, we evaluate prediction of functional protein sets derived from curated pathway annotations from Reactome[23]. Each positive example consists of the set of proteins assigned to a pathway. To construct negative examples, we generate "corrupted" pathways by replacing a fraction of proteins in a real pathway with randomly selected proteins, preserving pathway size while disrupting functional coherence. Because pathway sizes vary, we compute the median pathway size and pad or truncate each set to this fixed size. Protein embeddings within a set are aggregated using a multi-head attention pooling module, and the pooled representation is passed to a three-layer MLP to predict whether the set corresponds to a real pathway. The task is evaluated under a pathway-level split, ensuring that entire pathways in the test set are unseen during training.

Together, these tasks probe complementary aspects of the embedding space, providing a comprehensive evaluation of the biological information captured by the learned protein embeddings.

2.5.2 Stratified Evaluation by Protein Evidence Level

To evaluate how well the learned embeddings support inference across different evidence regimes, we stratify proteins according to the amount of available biological knowledge. We quantify evidence using the total degree of each protein in the heterogeneous training network, which reflects both annotation density and interaction support. All 18,049 proteins are ranked by degree and partitioned into four equal-sized quantiles. The lowest-degree quantile (Q1; 4,512 proteins) represents understudied proteins with sparse connectivity and limited prior characterization, whereas the highest-degree quantile (Q4; 4,513 proteins) corresponds to well-studied proteins with extensive interaction and annotation evidence. For each prediction task, we report performance metrics separately for understudied and well-studied proteins. This stratified evaluation enables a direct assessment of how effectively the embeddings generalize to low-evidence proteins, a setting of particular practical importance given that many biologically and clinically relevant proteins remain poorly characterized.

2.5.3 Obtaining embeddings from baseline methods

To benchmark the Gatsbi embeddings, we compare against representations from Pinnacle, the leading context-aware protein embedding approach[8]. Pinnacle embeddings were obtained from the publicly released model weights, which provide context-specific protein representations across multiple tissues and cell types. Because Pinnacle produces a separate vector for each protein–context pair, we derived a single per-protein embedding by mean pooling across all available contexts for that protein, following common practice for downstream evaluation. This pooled vector was then used directly as input to the same prediction pipelines applied to Gatsbi embeddings, ensuring that differences in performance reflect the information encoded in the embeddings rather than task-specific retraining.

Table 1 Structural statistics of individual data sources and the final heterogeneous protein network.

Data Source	# Nodes	# Edges	# Conn. Comp.	Avg. Degree
Protein–protein interaction	16,462	217,092	93	26.4
Expression	11,205	151,067	271	26.9
Tissue-context	17,617	1,207,151	39.8 (avg)	28.2 (avg)
Heterogeneous network	18,049	1,575,310	31(avg)	27.5 (avg)

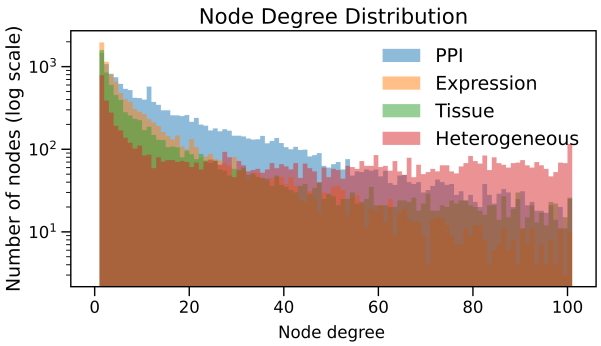

Figure 3 This figure shows a histogram of the edge degree for the nodes for each component data source and the combined heterogeneous graph. We see that the combined graph has the fewest low degree nodes, thus adding the most amount of information for the understudied proteins

3. Results

We report results for the experiments described in Sections 2.4 and 2.5, including performance of Gatsbi training under each data-splitting strategy and the effect of each split on all prediction tasks compared with baseline embedding methods.

3.1 Constructed heterogeneous graph captures context-specificity and adds more data for understudied proteins

Table 1 summarizes the number of nodes, edges, and connectivity statistics contributed by each individual data source as well as by the final heterogeneous graph. Figure 1 illustrates a representative subgraph of the heterogeneous network, highlighting the diverse edge types and context-specific relationships encoded between protein nodes. To understand how the distribution of edges in the understudied region, we plot a histogram of nodes at degree 0-100 in Figure 3. The heterogeneous graph contains fewer low-degree proteins than any single input network, indicating that proteins with limited experimental evidence in one data source are often supported by complementary evidence in others. As a result, the integrated graph offers improved context specificity and expanded information content for proteins that would otherwise remain poorly characterized. Figure 4 shows a histogram of the node degrees and the cutoffs of the well-studied (degree range 151-655) and understudied proteins (degree ¡ 38).

3.2 Gatsbi embeddings generalize consistently

We evaluate the generalization performance of Gatsbi under the data-splitting strategies (described in Section 2.4.1) edge and node (inductive) splits.

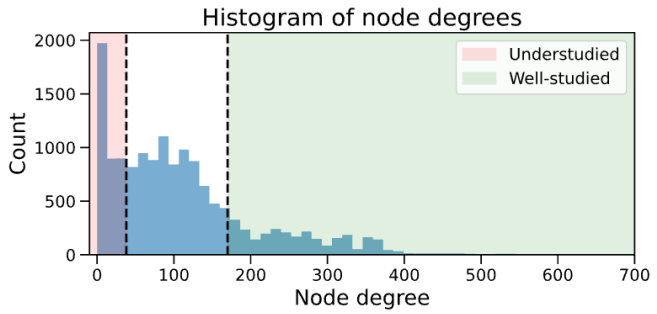

Figure 4 Using the method described in Section 2.5.2, the proteins are grouped into well-studied and understudied categories based on the node degree. The histogram shows the number of proteins at each node degree with the cutoffs delineated with vertical black lines. The understudied proteins are shaded in red, and they have a cutoff of degree 38. The well-studied proteins' degree range between 151-655. There are 3705 understudied proteins and 3641 well-studied proteins.

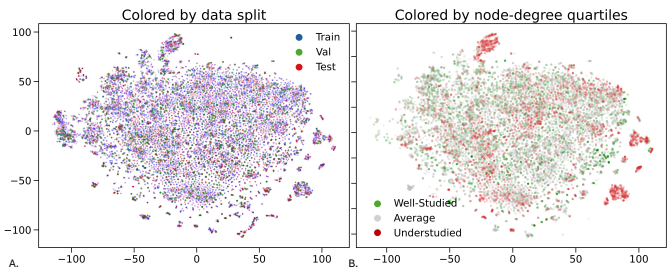

Figure 5 This figure shows the protein embedding space produced by the GAT network with the edge split. Panel A shows the embeddings colored by train, validation, and test sets; the sets of equally distributed across the space, showing that the embedding method can generalize across the space. Panel B shows the proteins with the well-studied proteins colored in green, the understudied proteins colored in red and the rest in grey (well-studied and understudied are defined by the quartile of node degree of the protein - described in Section 2.5.2). Visually, there are 3 clusters of mostly understudied proteins, and the remaining are close to a well-studied protein. The average cosine distance between an understudied protein and its nearest well-studied protein is 0.23.

Figure 5 provides a qualitative view of the learned embedding space using t-SNE projections for each data split. This figure visualizes the protein embedding space produced by the GAT model under the edge split. In Panel A, embeddings are colored by training, validation, and test membership; the absence of visible clustering by split indicates that the learned representations are not partition-specific and generalize smoothly across the space. Panel B colors proteins by evidence level, with well-studied proteins shown in green, understudied proteins in red, and the remainder in grey (defined by node-degree quartiles as described in Section 2.5.2). Understudied proteins are broadly distributed throughout the embedding space rather than confined to isolated regions, and most lie in proximity to well-studied proteins. Quantitatively, the average cosine distance between an understudied protein and its nearest well-studied neighbor is 0.23. Together, these observations suggest that GATSBI places proteins with limited prior evidence near informative reference proteins, supporting the ability of downstream models to annotate understudied proteins using shared contextual structure.

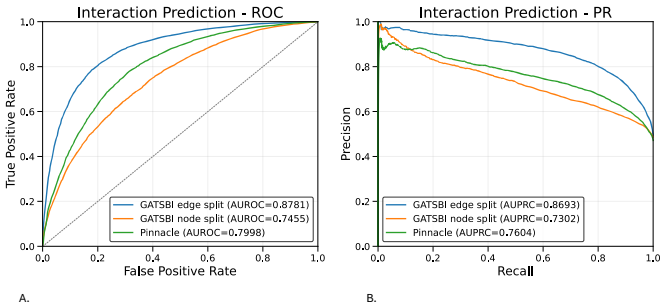

Figure 6 Interaction prediction performance. (A) ROC and (B) precision–recall curves show that GATSBI with the edge split outperforms PINNACLE, while the node split remains competitive despite evaluating generalization to unseen proteins.

3.3 GATSBI embeddings improve downstream prediction across tasks and splits

We next evaluate the quality of the learned protein embeddings on the four downstream prediction tasks described in Section 2.5.1: protein function prediction, interaction prediction, functional subgraph prediction, and disease association prediction. Tables 2-4 reports the performance of each task using standard metrics including AUC, accuracy, precision, recall, and F1 score, for embeddings obtained under the edge and node splits and the baseline embeddings.

For the interaction prediction task, GATSBI achieved the strongest performance when trained under the edge split, reaching an AUROC of 0.8781 and AUPRC of 0.8693, with an accuracy of 0.7942 and F1 score of 0.8073 (Table 2). These results exceed those obtained using PINNACLE embeddings (AUROC 0.7998, AUPRC 0.7604), indicating that the heterogeneous GAT training better captures functional relationships relevant for link recovery. As expected, performance under the more challenging inductive node split was lower (AUROC 0.7455, AUPRC 0.7302), yet GATSBI maintained high recall (0.8537), demonstrating that the model retains sensitivity to true interactions even for proteins unseen during training. The ROC and precision–recall curves (Figure 6) illustrate these trends, with the edge-split model showing consistently superior discrimination across operating points

Table 2 Performance comparison for the interaction prediction task

Method	Accuracy	AUROC	AUPRC	F1	Recall
Gatsbi-Edge*	0.802	0.882	0.871	0.807	0.865
	(.007)	(.005)	(.011)	(.007)	(.008)
GATSBI node**	0.647	0.740	0.729	0.706	0.859
	(.004)	(.005)	(.005)	(.009)	(.005)
PINNACLE	0.724	0.807	0.763	0.734	0.825
	(.010)	(.006)	(.006)	(.005)	(.009)

Values are mean on the first line and (95% confidence interval) on the second line.
* $p < 0.01$, ** $p < 0.001$ compared with PINNACLE.

For protein function prediction, GATSBI demonstrated strong performance under the inductive node split, achieving an AUROC of 0.6788 and AUPRC of 0.2306, outperforming PINNACLE embeddings (AUROC 0.6677, AUPRC 0.1973) (Table 3). The node-split model also attained higher accuracy (0.6315 vs. 0.6034) while maintaining competitive micro-F1 (0.3601 vs. 0.3679), indicating effective generalization to proteins unseen during training. In contrast, the edge-split model showed lower ranking performance (AUROC 0.5922,

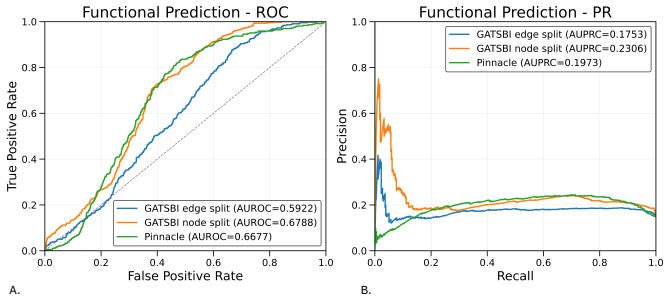

Figure 7 ROC (A) and precision–recall (B) curves for protein function prediction. The node-split model shows improved ranking performance over PINNACLE, while AUPRC values are low across methods due to label imbalance.

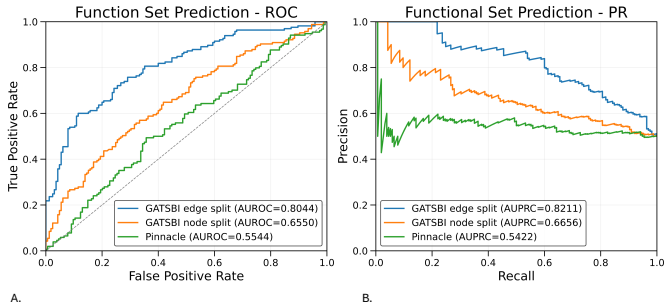

Figure 8 Functional set prediction performance. (A) ROC and (B) precision–recall curves comparing GATSBI under the edge and node splits with PINNACLE embeddings. The edge-split model achieves the strongest discrimination of protein sets, while the node split evaluates inductive generalization to proteins unseen during training. PINNACLE embeddings show high recall but reduced precision, highlighting the improved balance provided by GATSBI representations.

AUPRC 0.1753) despite very high recall (0.8764), suggesting that evaluation within the transductive regime favors sensitivity but leads to less precise label discrimination. The ROC and precision–recall curves (Figure 7) reflect these trends, with the node-split embeddings providing consistently better calibration across decision thresholds. The low AUPRC reflects extreme label imbalance, as each protein is associated with only a small subset of functional codes. We therefore report micro-averaged metrics, which aggregate over all label instances and avoid dominance by rare classes. These results indicate that learning under the inductive split yields representations that are more informative for functional annotation of understudied proteins than either the edge-split variant or existing pretrained embeddings.

Table 3 Performance comparison for function prediction (micro-averaged metrics).

Method	Accuracy (per-label)	AUROC (micro)	AUPRC (micro)	F1 (micro)	Recall (micro)
GATSBI edge*	0.423 (.007)	0.596 (.005)	0.177 (.011)	0.309 (.007)	0.871 (.008)
Gatsbi Node*	0.629 (.004)	0.673 (.005)	0.229 (.005)	0.363 (.009)	0.714 (.005)
PINNACLE	0.607 (.010)	0.674 (.006)	0.200 (.006)	0.367 (.005)	0.772 (.009)

Values are mean (first line) and 95% confidence interval (second line) over 10 runs.
* $p < 0.05$ compared with PINNACLE.

For the functional set prediction task, GATSBI trained with the edge split achieved the strongest overall performance, with AUROC 0.8044 and AUPRC 0.8211, substantially outperforming PINNACLE embeddings (AUROC 0.5544, AUPRC 0.5422) (Table 4). The edge-split model also attained the highest accuracy (0.7273) and F1 score (0.7472), indicating effective capture of shared functional context among protein groups. Under the inductive node split, performance decreased as expected (AUROC 0.6550, AUPRC 0.6656), yet GATSBI remained markedly better than PINNACLE while maintaining high recall (0.8970). PINNACLE showed very high recall (0.9416) but low precision, reflected in its reduced AUROC and AUPRC. The ROC and precision–recall curves (Figure 8) illustrate these trade-offs, with the edge-split model providing consistently superior discrimination across thresholds. These results demonstrate that GATSBI embeddings capture functional coherence among protein sets more effectively than existing pretrained representations, while the inductive model remains competitive in the challenging unseen-protein regime.

Table 4 Performance comparison for set prediction task.

Method	Accuracy	AUROC	AUPRC	F1	Recall
GATSBI edge*	0.734 (.007)	0.808 (.005)	0.823 (.011)	0.747 (.007)	0.800 (.008)
GATSBI node**	0.571 (.004)	0.649 (.005)	0.664 (.005)	0.680 (.009)	0.902 (.005)
PINNACLE	0.539 (.010)	0.561 (.006)	0.545 (.006)	0.669 (.005)	0.943 (.009)

Values are mean (first line) and 95% confidence interval (second line) over 10 runs.
* $p < 0.01$, ** $p < 0.001$ compared with PINNACLE

3.4 GATSBI performs higher than baseline method for understudied proteins

We further stratify evaluation results by protein degree quantiles, as described in Section 2.5.2. Performance stratified by protein evidence level confirms that GATSBI provides greater benefit for understudied proteins than existing pretrained embeddings (Table 5). For understudied proteins, the edge-split model achieved AUROC 0.781 and AUPRC 0.801, exceeding PINNACLE (AUROC 0.522, AUPRC 0.511), while maintaining a balanced F1 of 0.724 compared with PINNACLE's recall-driven behavior (F1 0.648, recall 0.928). The inductive node split also outperformed PINNACLE on ranking metrics (AUROC 0.632 vs. 0.522; AUPRC 0.641 vs. 0.511), demonstrating that GATSBI embeddings transfer more effectively to proteins lacking training evidence. Similar trends were observed for well-studied proteins, where the edge split reached AUROC 0.822 and AUPRC 0.836, again surpassing PINNACLE (AUROC 0.578, AUPRC 0.566). Quantitatively, the improvement over PINNACLE is largest in the understudied regime (ΔAUROC $= +0.259$; ΔAUPRC $= +0.290$) but remains substantial for well-studied proteins (ΔAUROC $= +0.244$; ΔAUPRC $= +0.270$), indicating consistent gains across evidence levels. Precision–recall behavior further highlights this difference: PINNACLE attains high recall at the expense of precision, whereas GATSBI achieves a more balanced operating point with higher F1. These results are consistent with the embedding-space analysis, where understudied proteins are positioned near informative well-studied neighbors, enabling effective knowledge transfer. The

stratified performance is visualized in Figure 9 A. The performance deltas (Figure 9 B) show that GATSBI provides the largest gains over PINNACLE for understudied proteins in interaction and functional set prediction, highlighting improved transfer to proteins with limited prior evidence. In contrast, for function prediction the improvement is similar across evidence levels, reflecting the inherently harder multiclass task where label imbalance, rather than protein coverage, constrains performance.

We qualitatively inspected false positives from the task models, particularly for understudied proteins, as these may represent relationships not yet reported. Figure 10 highlights two such predictions. Panel A shows a predicted interaction between the understudied proteins Protocadherin-15, involved in cochlear function, and Stereocilin, required for hair-cell stereocilia formation. Although this interaction is not documented in humans, a related association has been reported in mole-rats [24]. Panel B shows a predicted functional set in which amyloid beta precursor protein (APP) was added to a module containing neuroserpin, PDCD10, alpha-synuclein, and major prion proteins. These proteins share roles in neuronal development and cell-cycle regulation, and APP is a neuronal surface receptor that promotes synaptogenesis. While the original network lacked a direct APP–neuroserpin link, an association has been described by Yepes *et al.* [25].

Table 5 Performance separated by well-studied vs understudied for all 3 tasks

Split	Method	AUROC	AUPRC	F1	Recall
Under-studied	GATSBI edge split	0.781	0.801	0.724	0.786
	GATSBI node split	0.632	0.641	0.658	0.882
	PINNACLE	0.522	0.511	0.648	0.928
Well-studied	GATSBI edge split	0.822	0.836	0.765	0.823
	GATSBI node split	0.676	0.684	0.691	0.912
	PINNACLE	0.578	0.566	0.685	0.951

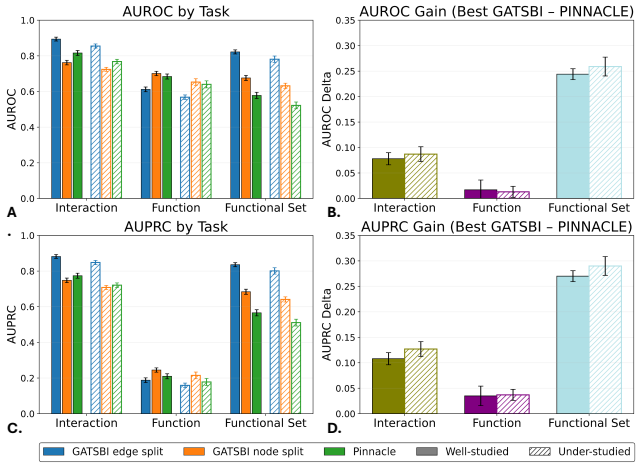

Figure 9 Performance of GATSBI and PINNACLE embeddings for well-studied and understudied proteins across three prediction tasks. (A) Stratified AUROC for interaction, function, and functional-set prediction using GATSBI (edge and node splits) and PINNACLE. GATSBI outperforms PINNACLE in both strata, with larger gains for understudied proteins. (B) AUROC delta (best GATSBI − PINNACLE) across tasks, with the largest improvements for understudied proteins in interaction and functional-set prediction. (C) Stratified AUPRC showing a consistent advantage of GATSBI, particularly for understudied proteins. (D) AUPRC delta, where gains are greatest for understudied proteins in interaction and functional-set prediction, while function prediction is primarily limited by multilabel imbalance.

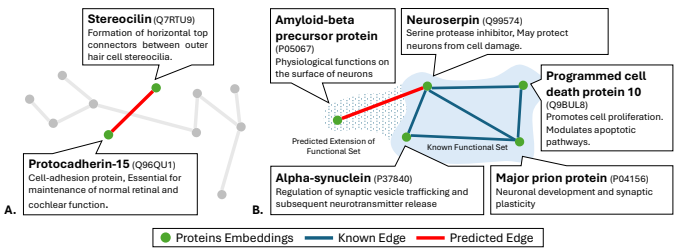

Figure 10 Visualization of two example predictions generated by the task models. Green nodes represent proteins, red edges denote model-predicted relationships, and blue edges indicate known associations from the input network. These cases illustrate how apparent false positives may reflect biologically plausible but previously unreported relationships, particularly for understudied proteins. (A) Predicted interaction between Protocadherin-15, involved in cochlear function, and Stereocilin, required for hair-cell stereocilia formation. (B) Predicted functional set in which amyloid beta precursor protein (APP) was added to a module containing neuroserpin, PDCD10, alpha-synuclein, and major prion proteins, all associated with neuronal development and cell-cycle regulation.

4. Discussion

This study demonstrates that the way protein embeddings are trained and evaluated has a decisive impact on their apparent utility for biological prediction tasks. By integrating heterogeneous evidence—interactions, co-expression, tissue-specific associations, and sequence features—into a single graph attention framework, GATSBI learns representations that consistently outperform existing pretrained embeddings across interaction, function, and functional set prediction. However, the most important finding is not only the absolute performance gain, but the difference in conclusions drawn under alternative evaluation protocols.

Evaluations conducted solely on well-studied proteins substantially overestimate real-world performance. PINNACLE is a pioneering effort in this space, introducing the first publicly available context-aware protein embeddings with distinct representations across tissues and cell types, and demonstrating the value of heterogeneous biological context for downstream prediction. Building on this foundation, GATSBI extends the approach with task-aligned training and evaluation strategies that better reflect the biological settings in which embeddings are used. When results are stratified by evidence level, the advantages of this extension become most pronounced for understudied proteins; PINNACLE tends to exhibit recall-driven behavior with limited precision, whereas GATSBI achieves a more balanced operating point. These findings suggest that heterogeneous message passing enables more effective transfer of contextual information from richly annotated regions to sparsely characterized proteins. The embedding-space analysis supports this interpretation, showing that understudied proteins are positioned near informative well-studied neighbors, providing a mechanistic basis for the improved downstream performance.

The comparison between edge and node splits further highlights that different biological questions require distinct evaluation strategies. The edge split measures recovery of missing relationships among known proteins and yields strong performance for interaction and set prediction. In contrast, the inductive node split better reflects scenarios in which entirely new proteins must be interpreted

without prior interaction evidence, as in functional annotation of newly discovered genes. The reversal of performance trends between these splits for certain tasks emphasizes that no single benchmark is sufficient; evaluation must be aligned with the intended application.

For function prediction, improvements were more modest and similar across evidence strata. This reflects the inherent difficulty of the multilabel setting, where extreme label imbalance limits achievable AUPRC regardless of embedding quality. The need for micro-averaged metrics and the persistent gap between AUROC and AUPRC indicate that future work should explore label-aware decoders and hierarchical modeling of functional codes, rather than relying solely on generic classifiers.

Several limitations remain. Although the node split reduces leakage through topology and sequence similarity, other forms of dependency—such as shared experimental biases across data sources—are harder to eliminate. Moreover, mean pooling of Pinnacle contexts, while necessary for fair comparison, may underutilize its conditional structure. Finally, our framework focuses on human proteins; extending to cross-species transfer and temporal contexts could further test generalization.

Overall, these results demonstrate that progress in protein representation learning depends as much on how embeddings are evaluated as on how they are constructed. By explicitly distinguishing transductive and inductive regimes and stratifying by evidence level, Gatsbi provides a more realistic assessment of practical utility, particularly for the long tail of understudied proteins that remain the central challenge of functional genomics.

Conflicts of interest

The authors declare that they have no competing interests.

Funding

GN is supported by NIH NLM 1F31LM014646. RBA supported by NIH GM153195, AI152960 and Burroughs Welcome Fund 1074128.

Data availability

Code and models for our experiments are available at https://github.com/Helix-Research-Lab/GATSBI-embedding

Author contributions statement

GN and RBA discussed the experiments. GN conducteed the experiments, analyzes the results, and wrote the manuscript. GN and RBA editted the manuscript.

References

1. R. Morris, K. A. Black, and E. J. Stollar, "Uncovering protein function: from classification to complexes," vol. 66, pp. 255–285.
2. X. Wang, D. Bo, C. Shi, S. Fan, Y. Ye, and P. S. Yu, "A survey on heterogeneous graph embedding: Methods, techniques, applications and sources," vol. 9, pp. 415–436.
3. C. Tran, S. Khadkikar, and A. Porollo, "Survey of protein sequence embedding models," vol. 24, p. 3775.
4. S. Ali, P. Chourasia, and M. Patterson, "When protein structure embedding meets large language models," vol. 15, p. 25.
5. Y. Xie, B. Yu, S. Lv, C. Zhang, G. Wang, and M. Gong, "A survey on heterogeneous network representation learning," vol. 116, p. 107936.
6. Y. Gu, S. Zheng, Q. Yin, R. Jiang, and J. Li, "REDDA: Integrating multiple biological relations to heterogeneous graph neural network for drug-disease association prediction," vol. 150, p. 106127.
7. A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph attention networks: A comprehensive review of methods and applications," vol. 16, p. 318.
8. M. M. Li, Y. Huang, M. Sumathipala, M. Q. Liang, A. Valdeolivas, A. N. Ananthakrishnan, K. Liao, D. Marbach, and M. Zitnik, "Contextual AI models for single-cell protein biology," vol. 21, pp. 1546–1557.
9. J. Bernett, D. B. Blumenthal, D. G. Grimm, F. Haselbeck, R. Joeres, O. V. Kalinina, and M. List, "Guiding questions to avoid data leakage in biological machine learning applications," vol. 21, pp. 1444–1453.
10. M. Milano, G. Agapito, and M. Cannataro, "Challenges and limitations of biological network analysis," vol. 11, p. 24.
11. E. Ursu, A. Minnegalieva, P. Rawat, M. Chernigovskaya, R. Tacutu, G. K. Sandve, P. A. Robert, and V. Greiff, "Training data composition determines machine learning generalization and biological rule discovery," vol. 7, pp. 1206–1219.
12. Y. Shao, H. Li, X. Gu, H. Yin, Y. Li, X. Miao, W. Zhang, B. Cui, and L. Chen, "Distributed graph neural network training: A survey," vol. 56, pp. 1–39.
13. N. Bordin, C. Dallago, M. Heinzinger, S. Kim, M. Littmann, C. Rauer, M. Steinegger, B. Rost, and C. Orengo, "Novel machine learning approaches revolutionize protein knowledge," vol. 48, pp. 345–359.
14. T. G. Brooks, N. F. Lahens, A. Mrčela, and G. R. Grant, "Challenges and best practices in omics benchmarking," vol. 25, pp. 326–339.
15. Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. Dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, and A. Rives, "Evolutionary-scale prediction of atomic-level protein structure with a language model," vol. 379, pp. 1123–1130.
16. D. Szklarczyk, K. Nastou, M. Koutrouli, R. Kirsch, F. Mehryary, R. Hachilif, D. Hu, M. E. Peluso, Q. Huang, T. Fang, N. T. Doncheva, S. Pyysalo, P. Bork, L. J. Jensen, and C. von Mering, "The STRING database in 2025: protein networks with directionality of regulation," vol. 53, pp. D730–D737.
17. T. Rout, A. Mohapatra, and M. Kar, "A systematic review of graph-based explorations of PPI networks: methods, resources, and best practices," vol. 13, p. 29.
18. A. Nogueira-Rodríguez, D. Glez-Peña, C. P. Vieira, J. Vieira, and H. López-Fernández, *PPI prediction from sequences via transfer learning on balanced but yet biased datasets: An open problem*, ser. Lecture Notes in Networks and Systems. Springer Nature Switzerland, pp. 31–40.
19. J. D. Montenegro, "Gene co-expression network analysis," vol. 2443, pp. 387–404.
20. R. S. G. Sealfon, C. L. Theesfeld, J. Funk, N. Sauerwald, A. K. Wong, and O. G. Troyanskaya, "HumanBase: an interactive AI platform for human biology," pp. 1–2.
21. R. Oughtred, J. Rust, C. Chang, B.-J. Breitkreutz, C. Stark, A. Willems, L. Boucher, G. Leung, N. Kolas, F. Zhang, S. Dolma, J. Coulombe-Huntington, A. Chatr-Aryamontri, K. Dolinski, and M. Tyers, "The BioGRID database: A comprehensive biomedical resource of curated protein, genetic, and chemical interactions," vol. 30, pp. 187–200.

22. A. G. McDonald and K. F. Tipton, “Enzyme nomenclature and classification: the state of the art,” vol. 290, pp. 2214–2231.
23. M. Milacic, D. Beavers, P. Conley, C. Gong, M. Gillespie, J. Griss, R. Haw, B. Jassal, L. Matthews, B. May, R. Petryszak, E. Ragueneau, K. Rothfels, C. Sevilla, V. Shamovsky, R. Stephan, K. Tiwari, T. Varusai, J. Weiser, A. Wright, G. Wu, L. Stein, H. Hermjakob, and P. D’Eustachio, “The reactome pathway knowledgebase 2024,” vol. 52, pp. D672–D678.
24. S. J. Pyott, M. van Tuinen, L. A. Screven, K. M. Schrode, J.-P. Bai, C. M. Barone, S. D. Price, A. Lysakowski, M. Sanderford, S. Kumar, J. Santos-Sacchi, A. M. Lauer, and T. J. Park, “Functional, morphological, and evolutionary characterization of hearing in subterranean, eusocial african mole-rats,” vol. 30, pp. 4329–4341.e4.
25. M. Yepes, Y. Woo, and C. Martin-Jimenez, “Plasminogen activators in neurovascular and neurodegenerative disorders,” vol. 22, p. 4380.