Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

PORTLAND PRESS

Review Article

Graph representation learning for structural proteomics

Romanos Fasoulis[1], Georgios Paliouras[2] and Lydia E. Kavraki[1]

[1]Department of Computer Science, Rice University, Houston, TX, U.S.A.; [2]Institute of Informatics and Telecommunications, NCSR Demokritos, Athens, Greece

Correspondence: Lydia E. Kavraki (kavraki@rice.edu)

The field of structural proteomics, which is focused on studying the structure–function relationship of proteins and protein complexes, is experiencing rapid growth. Since the early 2000s, structural databases such as the Protein Data Bank are storing increasing amounts of protein structural data, in addition to modeled structures becoming increasingly available. This, combined with the recent advances in graph-based machine-learning models, enables the use of protein structural data in predictive models, with the goal of creating tools that will advance our understanding of protein function. Similar to using graph learning tools to molecular graphs, which currently undergo rapid development, there is also an increasing trend in using graph learning approaches on protein structures. In this short review paper, we survey studies that use graph learning techniques on proteins, and examine their successes and shortcomings, while also discussing future directions.

Introduction

Proteins are the building blocks of all cells in our bodies. Although the DNA molecule holds all the information that is necessary for life, it is proteins that carry out what is coded in the genetic material [1]. As protein function is largely determined by its three-dimensional (3D) conformation, knowing the tertiary structure of a protein is a basic prerequisite for understanding its function [2]. While many specialized protein structural databases exist [3,4], the Protein Data Bank (PDB) is the de facto internationally recognized repository that holds determined experimentally 3D protein structures [5]. In the last two decades, we have seen a substantial increase in protein structures deposited in PDB [6], as well as an increase in its usage by scientists in the field [7]. Additionally, as a result of the success of Alphafold [8,9] models in predicting protein structures from their amino acid sequence, a large database was recently created, holding modeled structures of almost the entire human proteome [10].

Parallel to the increase in structural data in the field of biology, novel machine learning (ML) and deep learning (DL) approaches are being developed that can harness huge amounts of data to achieve high predictive performance [11,12]. In the last few years, increasing efforts have been made to expand DL techniques to the geometrical domain, in order to learn from complex structural data, particularly in tasks where the structural component is strong. As a result, the umbrella term *geometric deep learning* was created, encompassing these techniques [13], a subset of which comprises graph learning models that are used for modeling network relations, data-induced similarities, as well as 3D shapes [13]. Graph-based learning approaches have received praise and have achieved great results on benchmark network datasets, thus, encouraging researchers to employ those methods in different domains and applications. Graph based models have been used in recommender systems, social networks, materials research and others [14]. Graph learning models have also been employed in biological fields, with one of the most recent, bio-related successes involving molecular graph learning, a subfield in which graph learning models are used for predicting biochemical properties of molecules. Advances in this field resulted in both developing molecule-specific graph models that are more specialized in extracting/using molecular structure information [15] and advancing the graph learning field as a whole as well [16].

Received: 14 July 2021
Revised: 2 September 2021
Accepted: 13 September 2021

Version of Record published:
19 October 2021

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

Given the increase in protein structural data and the success of graph learning methods, it is natural for studies that are employing graph learning models in the structural proteomics field to emerge. The goals of this short review are to:

- Provide related work on protein graph-based representations.
- Introduce the graph representation learning (GRL) field, and explore its potential use to structural proteomics.
- Report studies in six different proteomics task categories where graph learning models have been successfully used.

Proteins as graphs

The graph representation of a protein structure collapses its 3D conformation into a graph, where now, the geometric information is incorporated within the graph connectivity, and not explicitly encoded in a coordinate system. Nodes in the graph can be defined at an amino acid level, where each node corresponds to a different amino acid in the protein sequence. Biochemical features for each node include can polarity, charge, hydrophobicity and others [17]. Protein graphs can also be defined at an atom level, where each node corresponds to an individual atom, and node features include the atom type and charge. The node-level representation of choice depends on the application. Atom-level protein graphs can, in principle, be more expressive, but the computational cost of working with a larger graph must be taken into account.

However, transforming a protein tertiary structure to a graph is far from trivial, as different topological and geometric information can be taken into account in the process. As such, different geometrical/biological methods arise for creating the graph. A simple way is to calculate all residue pairs distances, and introduce an edge between two residues if the distance is smaller than an empirically proposed cutoff δ. To avoid over-connectivity or exceeding sparsity, a k-nearest neighbors approach can also be used, where each node is connected to its k less distant neighbors. However, it is likely that this simple discretization does not capture accurately the geometrical structure of the underlying topological manifold [13]. Many studies have experimented with performing Delaunay triangulation (the dual graph of the Voronoi diagram), which is able to extract hierarchical molecular information about protein structure [18,19]. Such information though can result in a denser graph [20] (Figure 1C), possibly dilating useful information for the task at hand [21]. Computational tools have been developed that identify an edge in the graph as an existing intramolecular interaction [20], (i.e. hydrogen bonds, salt bridges, pi-cation bonds) providing additional useful biochemical information. A visual review of the methods discussed above can be seen in Figure 1.

Graph representations have been extensively used in protein analysis. The Gaussian network model [22,23] and the anisotropic network model [24,25] model the protein as an elastic network with 3D derived topological constraints, and have been pivotal in studying protein dynamics and flexibility. Graph-theory approaches have also been used for a variety of tasks, from protein structure identification [26], to side-chain prediction [27], among others (see [28,29] for graph theory methods in proteomics tasks). However, many of the aforementioned methods rely on theoretical techniques and models that are not specific to proteomics, while others incorporate empirical knowledge and constraints that may not be applicable to the task at hand. With the rise of graph learning and the increasing amount of protein structural data, an alternative approach becomes feasible; namely, one that relies on algorithms that learn the relevant structural information from graph data. In the following sections, we will describe the rising field of GRL and the way that knowledge can be learned from graph data in an end-to-end fashion.

Graph representation learning

Learning from graph structure

DL approaches have shown state-of-the-art results, partly due to their increased representational capacity [12], but also due to the inductive bias of the DL architectures themselves [30]. This is particularly true for convolutional neural networks (CNNs), which are very effective in processing image data, due to their locality and translational invariance [31]. Given their success, CNN architectures have been applied to more general, non grid-based data [13].

In this context, the emerging field of GRL seeks to represent graph data in such a way that it can be given as input to a standard neural network-like architecture for further downstream tasks. More formally, the idea is to

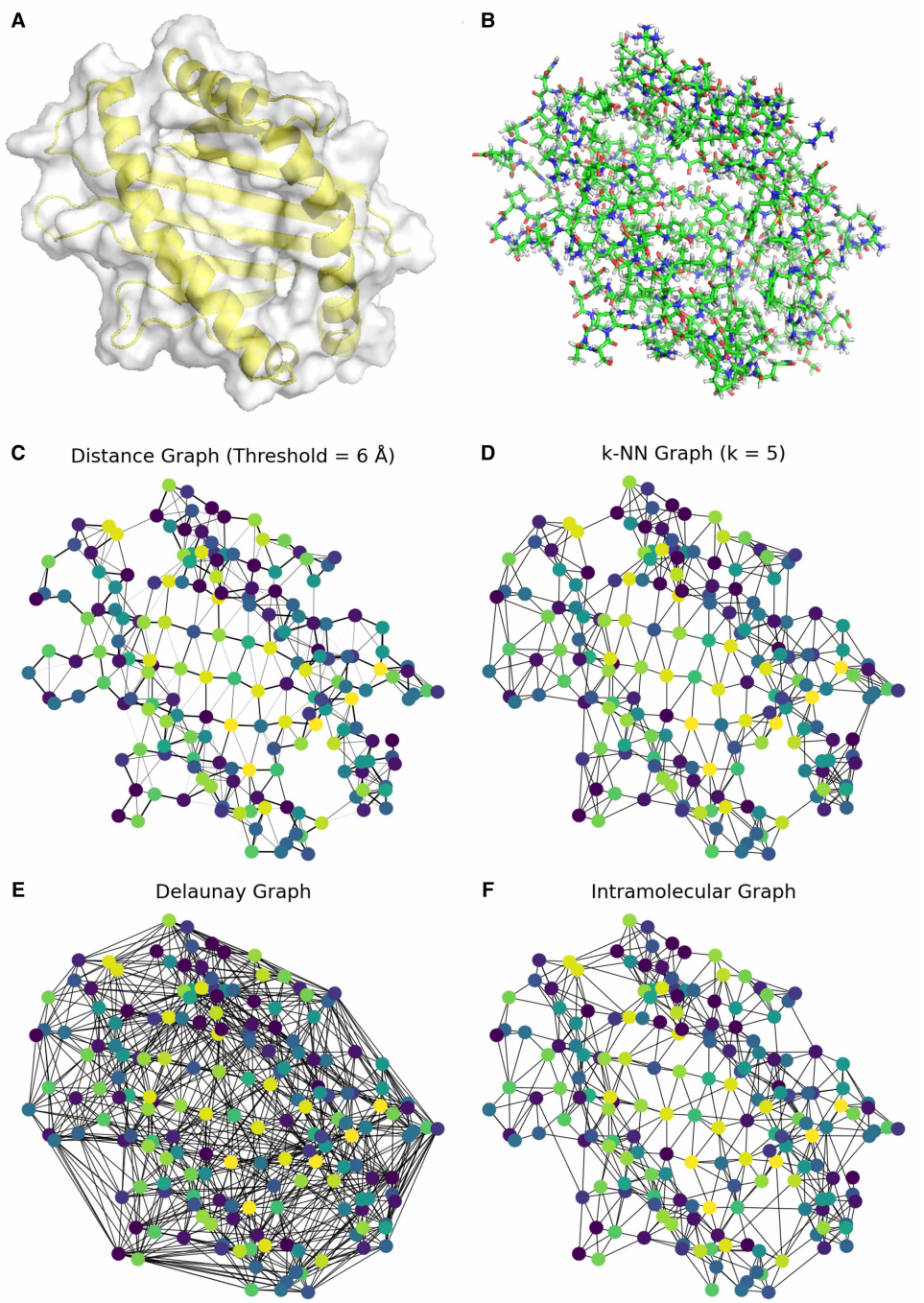

Fig. 1. Graphical representation of the binding interface of HLA-B*52:01 (PDB: 3W39, peptide removed).
Graphs are shown at an amino-acid level, each color corresponding to a different amino acid, and geometrical information is not expressed in 3D coordinates, but as relations/edges in the graph. (**A**) The 3D cartoon model, highlighted in yellow color, while the surface is shown in white color. (**B**) The 3D all-atom model, represented as sticks. (**C**) The distance graph, where each edge between two residues denotes that their actual distance is smaller than a cutoff δ (here equal to 6 Å). For calculating pairwise distances, α-carbon atoms are used as centroids. (**D**) The k-NN graph, where each residue is forced to connect to its $k = 5$ closest residue neighbors. (**E**) The Delaunay graph, created using Delaunay triangulation. (**F**) The intramolecular graph; each edge denotes a chemical bond (covalent/non-covalent).

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

learn a mapping function that compresses node, sub-graph, or entire graph information to a vector of fixed dimensions. This vector, containing not only node/graph attribute information, but also structural information stemming from the graph connectivity itself, can be used as a feature input for downstream tasks [32].

While embedding nodes and graphs goes far back [33,34], during the last few years, GRL has dominated the field, demonstrating state-of-the-art results in numerous graph-related tasks, such as node classification, link prediction, and graph classification [35]. One of the reasons for this is that more recent GRL approaches integrate the feature extraction and training processes in an end-to-end manner. In this way, they learn the appropriate graph embeddings for the task at hand, in contrast with methods that extract graph information in a task-agnostic way [36]. One of the most popular GRL methods, capable of achieving end-to-end learning, is the graph neural network (GNN).

Graph neural networks

GNNs appeared more than 10 years ago [37,38], and different variations were slowly surfacing [39,40]. However, GNNs became mainstream after the introduction of simplified operations, more specifically, after the seminal graph convolutional network (GCN) paper in 2017 [41]. Since then, the field has exploded, with different GNN models and extensions having various applications to many different domains [14].

Consider a protein graph G, comprised of residue nodes V and edges E, denoted as $G(V, E)$. The set of edges E will be different for different transformations of a protein structure to a graph (Figure 1). Each residue i has a set of biochemical features x_i. The set of neighbors of a residue i is denoted as N_i. GNNs use the following operator to calculate an embedding representation h_i of a node i at layer k of the network (see Figure 2 for a visual representation):

$$h_i^{(k)} = UPDATE(h_i^{(k-1)},\ AGGREGATE(\{h_u^{(k-1)} \mid u \in N_i\})), \quad h_i^{(0)} = x_i \tag{1}$$

What is achieved with the above operator for each node/residue is the transformation from a graph domain representation to a vector representation, while still retaining topological information. After the GNN operator is applied, a residue i will not be characterized only by its biochemical features x_i, but its embedding h_i will also contain biochemical information from its topological neighborhood. Many layers can be put together in series, so that information from distant neighbors can be obtained (Figure 2C). However, careful experimental selection of hyperparameters and number of GNN layers is crucial. Too much information from distant neighbors being aggregated into a single embedding can result in over-smoothing/over-squashing the important biological information that distinguishes each residue [21,42].

It is worth noting that equation (1) serves as a blueprint to design and create GNN models. There are many GNN variations [14,35], instantiating the UPDATE and AGGREGATE functions. In other words, they differ in how neighborhood information of a node i is aggregated, and how it is combined with its representation h_i [43]. For example, for a given node i, GCN's [41] AGGREGATE function averages and normalizes element-wise all neighbor embeddings (including the embedding of node i itself using self-connections). To UPDATE the new representation $h_i^{(k)}$ at layer k, the result from the AGGREGATE function passes through a simple one-layer feed forward neural network [12]. Hence, GCN calculates an embedding representation as follows:

$$h_i^{(k)} = \sigma\left(W^k \sum_{u \in N_i \cup i} \frac{1}{\sqrt{|N_u||N_i|}} h_u^{(k-1)}\right), \quad h_i^{(0)} = x_i \tag{2}$$

Another popular variant of GNNs is the graph attention network (GAT) [44], which is highly used in structural proteomics tasks (Table 1), due to its ability to select the important residue/atom neighbors for a given node. While GATs UPDATE function remains largely the same as GCNs, its AGGREGATE function differs from GCNs in that the averaging is weighted, with the weights $a_{u,i}$ being learned from data:

$$h_i^{(k)} = \sigma\left(W^k \sum_{u \in N_i \cup i} a_{u,i} h_u^{(k-1)}\right), \quad h_i^{(0)} = x_i \tag{3}$$

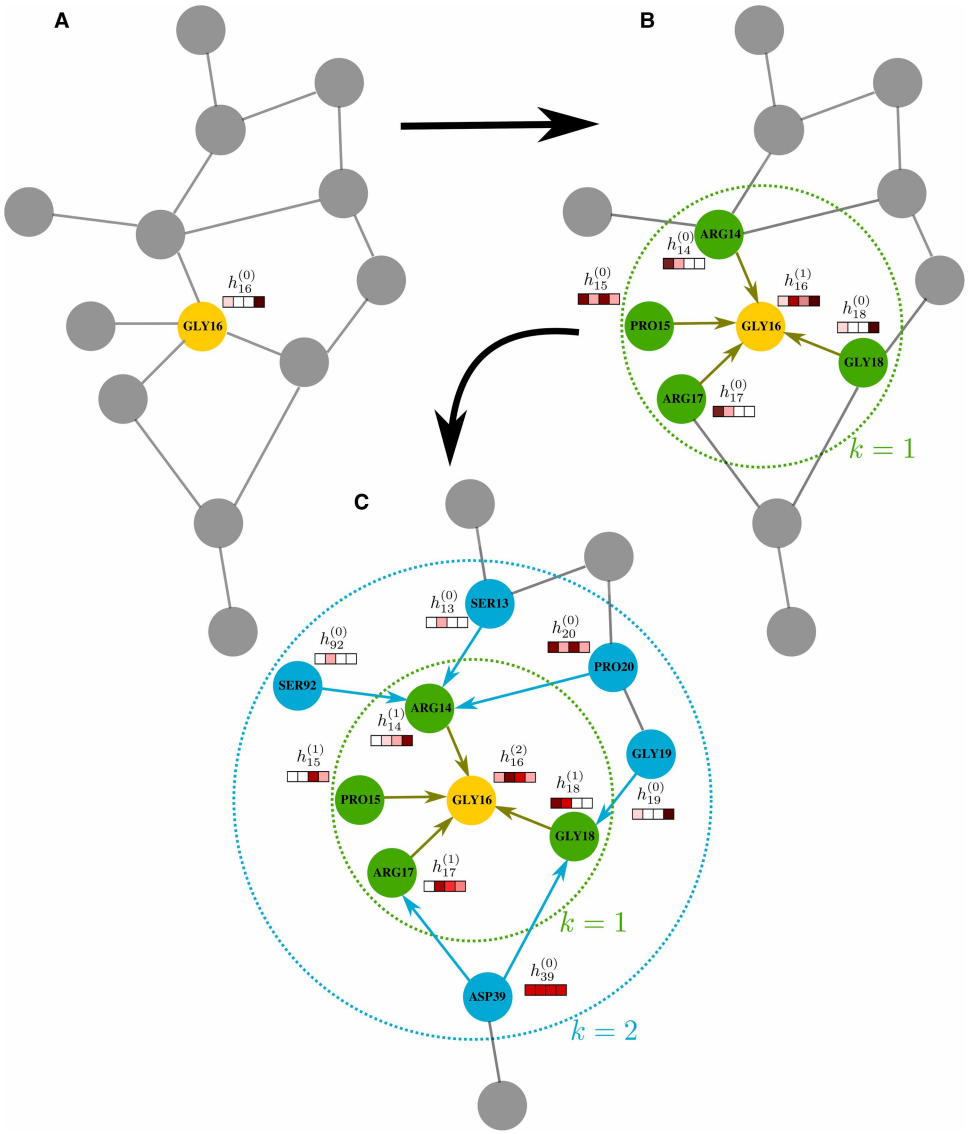

Fig. 2. Convolution operations performed on a protein subgraph.

(**A**) The residue of interest here is GLY16, and $h_{16}^{(0)} = x_{16}$ represents its biochemical features. (**B**) The first GNN layer. Biochemical features from residues belonging to GLY16's immediate neighbors are aggregated and transformed in order to calculate the embedding $h_{16}^{(1)}$, injecting topological information in the process. (**C**) Multiple convolutional operations can be used in series to take into account information from distant neighbors. This however can potentially cause *oversmoothing* on the GLY16 embedding [42].

There are numerous review papers on explaining and analyzing different GNN models, and presenting them in detail is out of the scope of this paper ([14,35] provide a review).

Protein topology has a central role in many biological processes. Therefore, GNNs, and the GRL field as a whole, being able to harness topological information, can potentially be a great tool in advancing the structural proteomics field. In the next few sections, we will examine GRL approaches that have been used for six proteomics-related tasks (Table 1).

Protein–ligand interaction

Protein–ligand interaction has been a well-studied problem. The underlying physicochemical mechanisms, i.e., binding kinetics, basic thermodynamic concepts, and the binding driving forces/factors have been extensively

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

Table 1. List of recent graph-learning-based methods for structural proteomics tasks

Category	Study	Graph type	Edge type	GNN layer type	Year	Reference
Protein–ligand prediction	GraphBAR	Atom graph (protein–ligand)	Distance edges (protein–ligand)	GCN-based	2021	[45]
	Lim et al.	Atom graph (protein–ligand) + atom graph (protein) + atom graph (ligand)	Distance edges (protein–ligand) + chemical bonds (protein) + chemical bonds (ligand)	GAT-based [44]	2019	[46]
	Torng et al.	Residue graph (protein) + atom graph (ligand)	Distance edges (protein) + chemical bonds (ligand)	GAE-based [47] + Duvenaud et al. [48]	2019	[49]
	DGraphDTA	Residue graph (protein) + atom graph (ligand)	Distance edges (protein) + chemical bonds (ligand)	GCN-based	2020	[50]
	GEFA	Residue graph (protein) + atom graph (ligand)	Distance edges (protein) + chemical bonds (ligand)	GCN-based + skip connections [51]	2020	[52]
Binding site identification	Fout et al.	Residue graph (proteins)	*k*-NN edges (proteins)	GCN-based + edge features	2017	[53]
	PECAN	Residue graph (proteins)	Distance edges (proteins)	GCN-based	2020	[54]
	PepNN	Residue graph (proteins)	*k*-NN edges (proteins)	Graph transformer [55]	2021	[56]
Docking scoring	EGCN	Residue graph (protein–ligand)	Distance edges (protein–ligand)	GCN-based + edge features	2020	[57]
	GNN-DOVE	Atom graph (protein–ligand) + atom graph (protein) + atom graph (ligand)	Distance edges (protein–ligand) + chemical bonds (protein) + chemical bonds (ligand)	GAT-based	2021	[58]
	InterPepRank	Residue graph (protein–ligand)	Distance edges (protein–ligand) + chemical bonds (protein–ligand)	Simonovsky et al. [59]	2020	[60]
Protein model quality assessment	GraphQA	Residue graph (protein)	Distance edges (protein) + covalent bonds (protein)	Graph Nets [31]	2021	[61]
	ProteinGCN	Atom graph (protein)	*k*-NN edges (protein)	Xie et al. [62]	2020	[63]
	VoroCNN	Atom graph (protein)	Voronota [64] edges (protein) + chemical bonds (protein)	GCN-based	2021	[65]
	S-GCN	Residue graph (protein)	Voronota edges (protein)	Custom	2020	[66]
Protein function prediction	DeepFRI	Residue graph (protein)	Distance edges (protein)	GAT-based	2021	[67]
	PersGNN	Residue graph (protein)	Distance edges (protein)	GCN-based	2020	[68]
	Gelman et al.	Residue graph (protein)	Distance edges (protein)	GCN-based	2021	[69]
Protein design	ProteinSolver	Residue graph (protein)	Distance edges (protein)	Wang et al. [70]	2020	[71]
	MimNet	Residue graph (protein)	Distance edges (protein)	GCN-based + U-net [72]	2021	[73]
	Ingraham et al.	Residue graph (protein)	*k*-NN edges (protein)	Graph transformer	2019	[74]

studied [75]. However, the way that those mechanisms intersect and contribute to the binding is very complex. Instead of trying to explain the interactions in depth, the increase in structural data allows graph-based methods to learn from data and provide accurate protein–ligand interaction predictions.

Studies have shown good drug-target affinity predictions when training a GNN on a sufficient amount of protein–ligand complex data (Figure 3A), and one such method is GraphBAR, a graph DL-based binding affinity prediction model [45]. GraphBAR represents the whole protein–ligand complex as a graph of connected atoms, that combines multiple adjacency matrices based on different distance measures, together with a feature matrix providing the molecular properties of the atoms. Graph convolutional operators are used to encode topological information of the complex, leading to superior performance over a CNN model for protein–ligand

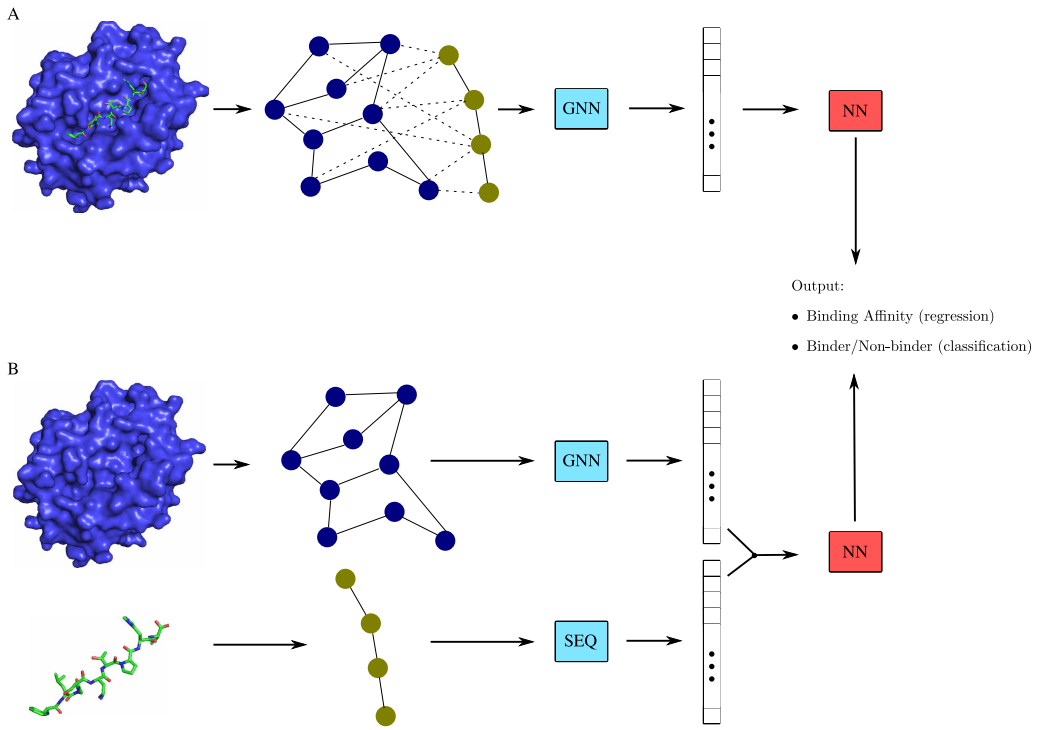

Fig. 3. Protein–ligand interaction studies using GNNs.

(**A**) Given a sufficient amount of protein–ligand structures, the graph of the whole complex can be given as an input to a GNN architecture. (**B**) Even when peptide–ligand complex data are not available, the protein complex and the ligand sequence can be processed as separate entities, retaining structural information during learning.

binding affinity prediction. In [46], a graph model is built that takes the atom graph representation of the protein–ligand binding pose to classify compounds as active or inactive. Two atom graphs are constructed, one including only covalent bonds, and one including both covalent/non-covalent bonds, and GAT [44] architectures are employed on both graphs to aggregate/differentiate between different molecular bonds. The paper reported improved performance compared with docking methods and other DL approaches.

GRL methods have been used even when protein–ligand complex data are not available. The corresponding studies treat the ligand and the protein as separate entities, with ligands being represented as small atom graphs, and proteins as large residue or atom-based graphs (Figure 3B). The authors of [49] use two separate GCNs to extract features from the protein binding pocket graph and the ligand atom graph. Their approach outperforms 3D-CNNs on virtual screening, while also producing interpretable results that indicate the contributions of each ligand atom and pocket residue to the classification decision. DGraphDTA [50] and GEFA [52] are methods that improve upon the earlier GraphDTA [76], which combines an atom-based graph representation for the ligand and a sequence representation for the protein. DGraphDTA first predicts the protein contact map based on the protein's amino acid sequence, and then it constructs a protein graph given the contact map. GCN architectures are used for both the protein residue graph and the ligand atom graph. GEFA introduces an early protein–ligand fusion approach, where, an attention-based mechanism connects the drug to the protein in a learnable fashion, and the unified protein–ligand representation passes through a GCN layer to predict drug-target affinities. Both studies achieve superior drug-target affinity predictions to GraphDTA and other purely sequence-based DL architectures like DeepDTA [77].

As the number of datasets and studies that GNN-based methods are tested is still low, it is unknown how well the methods discussed above generalize. Moreover, in the case where the protein–ligand complex is not given (Figure 3B), there is no sufficient evidence that the contributions of the ligand atoms presented in [49] or the protein–ligand connections learned in [52] correlate with actual structural protein–ligand data. However, it is evident that — whether there is enough protein–ligand complex data or not-using a graph representation

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

and applying GNN-based architectures can improve binding affinity predictions compared with CNN-like approaches and sequence-based models [46,50].

Binding site identification

In a protein interaction, it is not only valuable to know the binding affinity, but also, the components that determine the interaction, i.e. the binding sites. In one of the first studies to use GRL approaches to structural proteomics, a protein–protein interaction system based on multiple stacked layers of graph convolutions was developed [53]. Given a pair of amino-acid residues from two different proteins, the system predicts whether the amino-acids will interact. The proposed method was shown to outperform a simpler method not based on graph convolutions, indicating that the use of information from a residue's neighbors improves the accuracy of interface prediction. PECAN employs a similar type of framework, employing two GCNs and an attention module to assign each node of the primary structure a probability of belonging to the binding interface [54]. In a more recent work, the authors at [56] developed two different peptide-binding site identification models, one based on structure (PepNN-Struct) and one on sequence (PepNN-Seq). Similarly to [53], they showed that incorporating structure and encoding it as a graph improves performance on different test sets, showcasing the potential of GNN-based approaches to binding site identification.

Scoring for docking

The 3D conformation of a receptor–ligand complex can affect specific biologically related functions, such as driving the cellular immune response [78]. To this end, various computational approaches have been developed that predict the 3D conformation of a ligand to a receptor [79–81]. Docking consists of sampling candidate conformations of the ligand and scoring them. GRL approaches have been proposed for ranking candidate protein–peptide and protein–protein conformations (see Figure 4 for an illustration of a protein–peptide docking scoring system). InterPepRank [60] generates a large dataset of protein–peptide complex decoys using FFT-docking [82] and then trains a GCN to predict the ligand root-mean-square deviation of decoys through edge-conditioned graph convolutions. In the context of protein–protein scoring, EGCN [57] uses a deep graph learning model to rank protein–protein docking models. EGCN significantly improves the ranking for a CAPRI test set involving homology docking. Finally, similarly to [46], GNN-DOVE [58] builds two different graphs, one using only the covalent bonds of the protein–protein pair, and one that considers non-covalent, distance based connections. GNN-DOVE uses both to classify whether a decoy is correct or not.

Overall, GNNs seem to show promise in regards to scoring conformations. However, it is questionable how well receptor–ligand graphs with microscopic structural differences can be differentiated with a GNN for

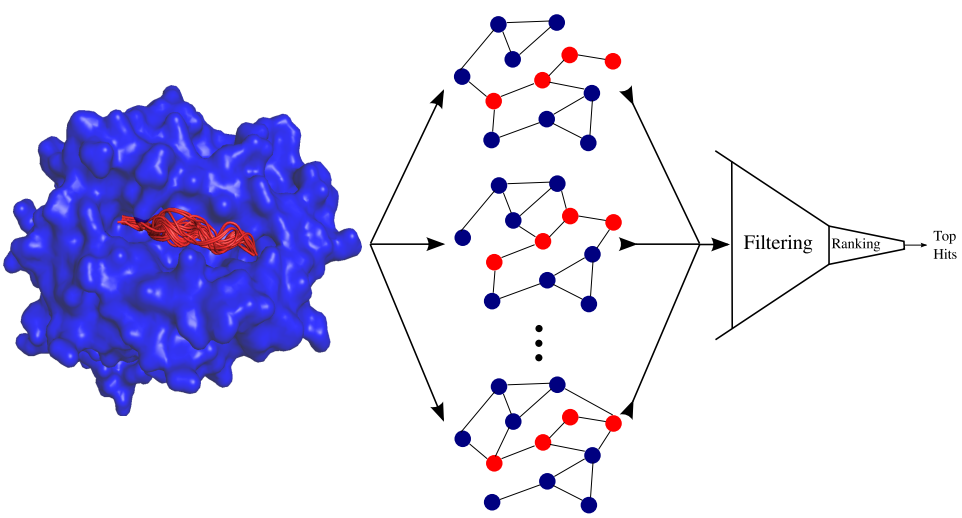

Fig. 4. Proof-of-concept pipeline for protein-peptide docking scoring using GNNs.

The filtering and ranking system, based on graph message passing GNN modules, takes graph conformations as input, and lists the top scoring ones.

scoring. Additionally, no extensive tests have been performed with ML-based scoring functions that use more fine-grained structural features than the whole receptor–ligand graph [83].

Protein model quality assessment

Predicting the 3D structure of a protein given its sequence is one of the most challenging problems in the field of computational biology [84]. Despite the undoubtedly great advances that Alphafold [8,9] has achieved in the recent critical assessment of protein structure prediction (CASP) challenges, the problem is still far from being solved, as the accuracy of predictions can still vary significantly [66]. Therefore, estimating the reliability of a modeled structure is very important. This task is known as protein model quality assessment (QA), and it is one of the sub-challenges of CASP [85]. What QA boils down to is scoring, both locally (per residue) and globally (whole complex), how reliable a modeled structure is, a very helpful metric when there is no experimental structure available.

In line with graph-based methods exploding in popularity, GNN-based approaches have also been tested in QA, with GraphQA [61], based on Graph Nets [31], being one of them. GraphQA, using a residue graph representation of the protein, achieves increased performance on both local and global quality assessment tasks in comparison with other QA methods [61]. ProteinGCN [63] is similar to GraphQA, but differs in the representation of the protein graph -which is done at an atom-level- and the convolutional operators [62]. While ProteinGCN has not been compared direclty to GraphQA, increased performance to state-of-the-art QA methods is reported [63]. Instead of computing the protein graph through a residue distance like in GraphQA or through a *k*-nearest-neighbor approach like in ProteinGCN, VoroCNN [65] builds a graph through the Voronoi 3D tessellation of a protein 3D model. Using the resulting graph, a GNN predicts local qualities of 3D protein folds. Finally, S-GCN [66] is a different convolution operator based on spherical harmonics, which is more suited in exploiting geometrical and topological information of the protein graph, achieving superior performance to state-of-the-art QA methods.

The abundance of different graph methods, as well as the increased performance gains in comparison to other methods, indicate that GNN-based architectures are well suited to the QA task. As the methods presented above employ different graph convolution architectures and graph representations, future work includes benchmarking those methods on different datasets to further investigate which graph representations and architectures work best.

Protein function prediction

Protein function prediction is a challenging task that is approached by various sequence-based and structural-based methods [86]. However, the fact that the function of a protein is intrinsically related to its 3D conformation (more than to its primary sequence) motivates the use of structure in predicting protein function. Many studies have used a graph representation of the protein, indicative of its geometry and topology, to predict protein function. DeepFRI [67] uses GNNs to predict protein function, by leveraging both sequence features extracted from a protein language model and protein structures. DeepFRI achieved better performance than its CNN predecessor DeepGO [87] which mainly uses protein sequence data. PersGNN [68] is another hybrid model that uses GNN operators, combined with persistent homology [88] to outperform simple neural-networks and vanilla GNNs on protein function prediction. The extensive benchmarking of GNN models in ENZYMES [89], PROTEINS [90] and D&D (Dobson and Doig) [91] datasets demonstrates the good performance of GNN models on a range of state-of-the-art protein function prediction tasks (see supplementary material of [92]).

Despite the good results obtained so far, it is still not clear if the graph representation of a structure can aid in predicting protein function. In particular, the study in [69] shows that introducing topological information and applying a GCN architecture on the protein graph may not improve sequence-to-function predictions. Moreover, one experiment showed that using different versions of the original protein graph, namely, a shuffled graph (residues at different locations), a disconnected graph (no edges), a fully connected graph (all residues connected) and a sequential-graph (covalent bonds only) produces similar results, indicating that the structural protein graph priors do not affect the final predictions. Therefore, more in-depth studies are needed to experimentally validate the use of graph learning in protein function prediction.

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

Protein design

Protein design seeks to support the manual engineering of proteins for specific functions, without relying on evolutionary mechanisms [93]. More formally, given a protein structure, the goal is to find a plausible underlying amino-acid sequence (inverse protein-folding problem) [74]. Advances in protein structure prediction [8,9] have motivated corresponding advances in protein design. Taking advantage of protein structural data, GRL methods have been used to predict effective protein sequences. An example of such a method is ProteinSolver [71], which formulates protein design as a constraint satisfaction problem. To overcome computational complexity, it employs a GNN architecture to encode the constraint graph, where nodes are the amino-acids in the protein sequence, and the edges are the constraints that stem from the protein contact map. The authors show that ProteinSolver can be successful in generating novel protein sequences for a predetermined fold. MimNet improves on ProteinSolver on all CASP 7-12 challenges by using a specialized GCN-based architecture that solves the protein structure and protein design problems in tandem [73]. This is achieved by it's bi-directional neural network architecture, which allows it to train for both protein folding and protein design simultaneously, doubling the amount of data available. Lastly, the *Structured Transformer* introduced in [74], inspired by machine translation tasks, 'translates' an input protein structure to a sequence profile. Among other results, it achieves better per-residue perplexities than purely sequence models, showcasing the power of GNN-based protein designed models that are conditioned on the structure.

GNNs have also shown potential in aiding protein design in other ways. In [94], a variation of GNNs [95] is employed to learn from molecular dynamics simulation data to infer protein allostery. The model proposed correctly infers the pathways that can mediate the allosteric communications between two binding sites. These results can aid protein design by mutating the amino acids in the allosteric pathways, in order to change protein function. This, combined with the ability to use inverse architectures to guide protein design [69,73], showcases the existing fertile ground for future protein design studies.

Conclusion

This paper examines the potential of GRL to incorporate protein structures in various biological problems. The increasing availability of structural datasets and the recent explosion of work in the GRL field have increased the potential of exploiting structural priors in downstream proteomics tasks. Such structural information could be the primary reason why graph-based models edge out their sequence-based counterparts in regards to the studies discussed.

The GRL field was introduced essentially in the last 5 years, and, ever since then, it has made a huge impact on multiple scientific fields. Given the preliminary results from proteomics-related studies that employ GRL, its use to bioinformatics and computational biology is expected to inspire much structural proteomics research in upcoming years.

Summary

- The increased availability of protein structural data has enabled the use of graph deep learning approaches in structural proteomics.
- GRL approaches have already been successfully employed in a multitude of tasks in many different domains/fields.
- In structural proteomics, GRL approaches have been used in protein–ligand interaction, protein function prediction and protein design, among others.
- Results from the studies discussed indicate that there is potential in using GRL methods together with ever-increasing protein structure data for a multitude of structural proteomics tasks.

PORTLAND PRESS

Competing Interests

The authors declare that there are no competing interests associated with the manuscript.

Funding

R.F. and L.E.K. are supported in part by NIH U01CA258512.

Acknowledgements

The authors thank Mauricio Menegatti Rigo and Anja Conev for their helpful discussions and comments on early drafts.

Abbreviations

CNNs, convolutional neural networks; DL, deep learning; GAT, graph attention network; GCN, graph convolutional network; GRL, graph representation learning; ML, machine learning; PDB, Protein Data Bank.

References

1 A. Breda, N.F. Valadares, O. Norberto de Souza and R.C. Garratt (2006) Protein structure, modelling and applications. In *Bioinformatics in Tropical Disease Research A Practical and Case-Study Approach*, (Gruber, A., Durham, A.M., Huynh, C. and del Portillo, H.A., eds), pp. 266–290, National Center for Biotechnology Information (US), Bethesda, MD
2 M. Bhasin and G.P.S. Raghava (2006) 8 - Computational methods in genome research. In *Applied Mycology and Biotechnology* (Arora, D.K., Berka, R. M. and Singh, G.B., eds), vol. 6, pp. 179–207, Elsevier, Amsterdam
3 S. Pawlicki, A. Le Béchec and C. Delamarche (2008) AMYPdb: a database dedicated to amyloid precursor proteins. *BMC Bioinform.* **9**, 273 https://doi.org/10.1186/1471-2105-9-273
4 S. Perez, A. Sarkar, A. Rivet, C. Breton and A. Imberty (2015) Glyco3D: a portal for structural glycosciences. *Methods Mol. Biol.* **1273**, 241–258 https://doi.org/10.1007/978-1-4939-2343-4
5 H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat and H. Weissig et al. (2000) The protein data bank. *Nucleic Acids Res.* **28**, 235–242 https://doi.org/10.1093/nar/28.1.235
6 P. Agrawal, S. Patiyal, R. Kumar, V. Kumar, H. Singh and P.K. Raghav et al. (2019) ccPDB 2.0: an updated version of datasets created and compiled from Protein Data Bank. *Database* **2019**, Bay142 https://doi.org/10.1093/database/bay142
7 C. Markosian, L. Di Costanzo, M. Sekharan, C. Shao, S.K. Burley and C. Zardecki (2018) Analysis of impact metrics for the protein data bank. *Sci. Data* **5**, 180212 https://doi.org/10.1038/sdata.2018.212
8 A.W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre and T. Green et al. (2020) Improved protein structure prediction using potentials from deep learning. *Nature* **577**, 706–710 https://doi.org/10.1038/s41586-019-1923-7
9 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov and O. Ronneberger et al. (2021) Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 https://doi.org/10.1038/s41586-021-03819-2
10 K. Tunyasuvunakool, J. Adler, Z. Wu, T. Green, M. Zielinski and A. Žídek et al. (2021) Highly accurate protein structure prediction for the human proteome. *Nature* **596**, 590–596 https://doi.org/10.1038/s41586-021-03828-1
11 I. Sarker (2021) Machine learning: algorithms, real-world applications and research directions. *SN Comput. Sci.* **2**, 160 https://doi.org/10.1007/s42979-021-00592-x
12 F. Emmert-Streib, Z. Yang, H. Feng, S. Tripathi and M. Dehmer (2020) An introductory review of deep learning for prediction models with big data. *Front. Artif. Intell.* **3**, 4 https://doi.org/10.3389/frai.2020.00004
13 M.M. Bronstein, J. Bruna, Y. LeCun, A. Szlam and P. Vandergheynst (2017) Geometric deep learning going beyond euclidean data. *IEEE Signal Process. Mag.* **34**, 18–42 https://doi.org/10.1109/MSP.79
14 Y. Zhou, H. Zheng and X. Huang (2021) Graph neural networks: taxonomy, advances and trends. *CoRR* **abs/2012.08752**
15 O. Wieder, S. Kohlbacher, M. Kuenemann, A. Garon, P. Ducrot and T. Seidel et al. (2020) A compact review of molecular property prediction with graph neural networks. *Drug Discov. Today Technol.* **59**, 12 https://doi.org/10.1016/j.ddtec.2020.11.009
16 J. Gilmer, S.S. Schoenholz, P.F. Riley, O. Vinyals and G.E. Dahl (2017) Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (D. Precup and Y.W. Teh, eds.), ICML'17, pp. 1263–1272, JMLR.org, Sydney, NSW, Australia, August 2017
17 S. Kawashima and M. Kanehisa (2000) AAindex: amino acid index database. *Nucleic Acids Res.* **28**, 374 https://doi.org/10.1093/nar/28.1.374
18 T. Taylor and I. Vaisman (2006) Graph theoretic properties of networks formed by the Delaunay tessellation of protein structures. *Phys. Rev. E* **73**, 041925 https://doi.org/10.1103/PhysRevE.73.041925
19 W. Zhou and H. Yan (2012) Alpha shape and Delaunay triangulation in studies of protein-related interactions. *Brief. Bioinformatics* **15**, 54–64 https://doi.org/10.1093/bib/bbs077
20 A.R. Jamasb, P. Lió and T.L. Blundell (2020) Graphein - a python library for geometric deep learning and network analysis on protein structures. *bioRxiv*
21 U. Alon and E. Yahav (2020) On the bottleneck of graph neural networks and its practical implications, *CoRR*, http://arxiv.org/abs/2006.05205
22 I. Bahar, A.R. Atilgan and B. Erman (1997) Direct evaluation of thermal fluctuations in proteins using a single-parameter harmonic potential. *Fold. Des.* **2**, 173–181 https://doi.org/10.1016/S1359-0278(97)00024-2
23 T. Haliloglu, I. Bahar and B. Erman (1997) Gaussian dynamics of folded proteins. *Phys. Rev. Lett.* **79**, 3090–3093 https://doi.org/10.1103/PhysRevLett.79.3090
24 A.R. Atilgan, S.R. Durell, R.L. Jernigan, M.C. Demirel, O. Keskin and I. Bahar (2001) Anisotropy of fluctuation dynamics of proteins with an elastic network model. *Biophys. J.* **80**, 505–515 https://doi.org/10.1016/S0006-3495(01)76033-X

25 P. Doruker, A. Atilgan and I. Bahar (2000) Dynamics of proteins predicted by molecular dynamics simulations and analytical approaches application to *alpha*-amylase inhibitor. *Proteins: Struct. Funct. Bioinformatics* **40**, 512–524 https://doi.org/10.1002/(ISSN)1097-0134
26 Y. Yan, S. Zhang and F.X. Wu (2011) Applications of graph theory in protein structure identification. *Proteome Sci.* **9** (Suppl 1), S17 https://doi.org/10.1186/1477-5956-9-S1-S17
27 A. Canutescu, A. Shelenkov and R.L. Dunbrack (2003) A graph-theory algorithm for rapid protein side-chain prediction. *Protein Sci.* **12**, 2001–2014 https://doi.org/10.1110/ps.03154503
28 K. Xia and G.W. Wei (2016) A review of geometric, topological and graph theory apparatuses for the modeling and analysis of biomolecular data. *CoRR* **abs/1612.01735**
29 S. Vishveshwara, K. Břinda and N. Kannan (2002) Protein structure insights from graph theory. *J. Theor. Comput. Chem.* **1**, 187–211 https://doi.org/10.1142/S0219633602000117
30 A. Goyal and Y. Bengio (2020) Inductive biases for deep learning of higher-level cognition. *CoRR* **abs/2011.15091** https://dblp.org/rec/journals/corr/abs-2011-15091.bib
31 P.W. Battaglia, J.B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V.F. Zambaldi and M. Malinowski et al. (2018) Relational inductive biases, deep learning, and graph networks. *CoRR* **abs/1806.01261** http://arxiv.org/abs/1806.01261
32 W.L. Hamilton, R. Ying and J. Leskovec (2017) Representation learning on graphs: methods and applications. *IEEE Data Eng. Bull.* **40**, 52–74
33 B. Luo, R.C. Wilson and E.R. Hancock (2003) Spectral embedding of graphs. *Pattern Recognit.* **36**, 2213–2230 https://doi.org/10.1016/S0031-3203(03)00084-0
34 D. Cai, X. He, J. Han and T.S. Huang (2011) Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**, 1548–1560 https://doi.org/10.1109/TPAMI.2010.231
35 Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang and P.S. Yu (2021) A comprehensive survey on graph neural networks. *IEEE Trans. Neural Netw.* **32**, 4–24 https://doi.org/10.1109/TNNLS.5962385
36 D.D. Nguyen and G. Wei (2019) AGL-score: algebraic graph learning score for protein–ligand binding scoring, ranking, docking, and screening. *J. Chem. Inf. Model.* **59**, 3291–3304 https://doi.org/10.1021/acs.jcim.9b00334
37 M. Gori, G. Monfardini and F. Scarselli (2005) A new model for earning in graph domains. In *Proceedings of the International Joint Conference on Neural Networks*, Vol. 2, pp. 729–734, IEEE, Montreal, QC, Canada, August 2005
38 F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner and G. Monfardini (2009) The graph neural network model. *IEEE Trans. Neural Netw.* **20**, 61–80 https://doi.org/10.1109/TNN.2008.2005605
39 J. Bruna, W. Zaremba, A. Szlam and Y. Lecun (2014) Spectral networks and locally connected networks on graphs. English (US). In *International Conference on Learning Representations (ICLR2014)*, CBLS, Banff, AB, Canada, April 2014 http://arxiv.org/abs/1312.6203
40 M. Defferrard, X. Bresson and P. Vandergheynst (2016) Convolutional neural networks on graphs with fast localized spectral filtering, *CoRR*, http://arxiv.org/abs/1606.09375
41 T.N. Kipf and M. Welling (2017) Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, OpenReview.net, Toulon, France, April 2017
42 C. Cai and Y. Wang (2020) A note on over-smoothing for graph neural networks. *CoRR* **abs/2006.13318** https://arxiv.org/abs/2006.13318
43 W.L. Hamilton (2020) *Graph Representation Lerning*, Synthesis Lectures on Artificial Intelligence and Machine Learning, Vol. 14.3, pp. 1–159, Morgan and Claypool, New York, NY, USA
44 P. Velivcković, G. Cucurull, A. Casanova, A. Romero, P. Lió and Y. Bengio (2018) Graph attention networks. In *International Conference on Learning Representations*, OpenReview.net https://openreview.net/forum?id=rJXMpikCZ
45 J. Son and D. Kim (2021) Development of a graph convolutional neural network model for efficient prediction of protein–ligand binding affinities. *PLoS ONE* **16**, e0249404 https://doi.org/10.1371/journal.pone.0249404
46 J. Lim, S. Ryu, K. Park, Y. Choe, J. Ham and W. Kim (2019) Predicting drug-target interaction using a novel graph neural network with 3D structure-embedded graph representation. *J. Chem. Inf. Model.* **59**, 3981–3988 https://doi.org/10.1021/acs.jcim.9b00387
47 T.N. Kipf and M. Welling (2016) Variational graph auto-encoders. *CoRR* **abs/1611.07308** http://arxiv.org/abs/1611.07308
48 D.K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel and A. Aspuru-Guzik et al. (2015) Convolutional networks on graphs for learning molecular fingerprints. In *Advances in Neural Information Processing Systems* (Cortes, C., Lawrence, N., Lee, D., Sugiyama, M. and Garnett. R., eds), vol. 28, Curran Associates, Inc., Red Hook, NY
49 W. Torng and R. Altman (2019) Graph convolutional neural networks for predicting drug-target interactions. *J. Chem. Inf. Model.* **59**, 4131–4149 https://doi.org/10.1021/acs.jcim.9b00628
50 M. Jiang, Z. Li, S. Zhang, S. Wang, X. Wang and Q. Yuan et al. (2020) Drug-target affinity prediction using graph neural network and contact maps. *RSC Adv.* **10**, 20701–20712 https://doi.org/10.1039/D0RA02297G
51 T. Minh Le, V. Le, S. Venkatesh and T. Tran (2020) Dynamic language binding in relational visual reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20* (Bessiere, C., ed.), Main track, pp. 818–824, International Joint Conferences on Artificial Intelligence Organization, Yokohama, Japan
52 T. Nguyen, T. Nguyen, T. Le and T. Tran (2021) GEFA early fusion approach in drug-target affinity prediction. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* **abs/2009.12146** https://doi.org/10.1109/TCBB.2021.3094217
53 A. Fout, J. Byrd, B. Shariat and A. Ben-Hur (2017) Protein interface prediction using graph convolutional networks. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17*, pp. 6533–6542, Curran Associates Inc., Red Hook, NY, USA
54 S. Pittala and C. Bailey-Kellogg (2020) Learning context-aware structural representations to predict antigen and antibody binding interfaces. *Bioinformatics* **36**, 3996–4003 https://doi.org/10.1093/bioinformatics/btaa263
55 V.P. Dwivedi and X. Bresson (2020) A generalization of transformer networks to graphs. CoRR
56 O. Abdin, H. Wen and P.M. Kim (2021) PepNN: a deep attention model for the identification of peptide binding sites. *bioRxiv*
57 Y. Cao and Y. Shen (2020) Energy-based graph convolutional networks for scoring protein docking models. *Proteins: Struct. Funct. Bioinformatics* **88**, 1091–1099 https://doi.org/10.1002/prot.v88.8

58 X. Wang, S.T. Flannery and D. Kihara (2021) Protein docking model evaluation by graph neural networks. *Front. Mol. Biosci.* **8**, 647915 https://doi.org/10.3389/fmolb.2021.647915
59 M. Simonovsky and N. Komodakis (2017) Dynamic Edge-Conditioned Filters in Convolutional Neural Networks on Graphs. In *2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 29–38, IEEE https://doi.org/10.1109/CVPR.2017.11
60 I. Johansson-Åkhe, C. Mirabello and B. Wallner (2020) InterPepRank: assessment of docked peptide conformations by a deep graph network. *bioRxiv*
61 F. Baldassarre, D. Menéndez Hurtado, A. Elofsson and H. Azizpour (2020) GraphQA: protein model quality assessment using graph convolutional networks. *Bioinformatics* **37**, 360–366 https://doi.org/10.1093/bioinformatics/btaa714
62 T. Xie and J.C. Grossman (2018) Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Phys. Rev. Lett.* **120**, 145301 https://doi.org/10.1103/PhysRevLett.120.145301
63 S. Sanyal, I. Anishchenko, A. Dagar, D. Baker and P. Talukdar (2020) ProteinGCN: protein model quality assessment using graph convolutional networks. *bioRxiv*
64 K. Olechnovič and Č. Venclovas (2014) Voronota: a fast and reliable tool for computing the vertices of the voronoi diagram of atomic balls. *J. Comput. Chem.* **35**, 672–681 https://doi.org/10.1002/jcc.v35.8
65 I. Igashov, K. Olechnovic and M. Kadukova (2020) VoroCNN: deep convolutional neural network built on 3D Voronoi tessellation of protein structures. *bioRxiv*
66 I. Igashov, N. Pavlichenko and S. Grudinin (2021) Spherical convolutions on molecular graphs for protein model quality assessment. *Mach. learn.: sci. technol.* **2**, 045005 https://doi.org/10.1088/2632-2153/abf856
67 V. Gligorijevic, P.D. Renfrew, T. Kosciolek, J.K. Leman, K. Cho and T. Vatanen et al. (2019) Structure-based function prediction using graph convolutional networks. *bioRxiv*
68 N. Swenson, A. Krishnapriyan, A. Buluc, D. Morozov and K. Yelick et al. (2020) PersGNN: applying topological data analysis and geometric deep learning to structure-based protein function prediction. *CoRR* **abs/2010.16027** https://arxiv.org/abs/2010.16027
69 S. Gelman, S.A. Fahlberg, P. Heinzelman, P.A. Romero and A. Gitter (2021) Neural networks to learn protein sequence-function relationships from deep mutational scanning data. *bioRxiv*
70 Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein and J.M. Solomon (2019) Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph* **38**, 1–12 https://doi.org/10.1145/3326362
71 A. Strokach, D. Becerra, C. Corbi-Verge, A. Perez-Riba and P.M. Kim (2020) Fast and flexible protein design using deep graph neural networks. *Cell. Syst.* **11**, 402.e4–411.e4 https://doi.org/10.1016/j.cels.2020.08.016
72 O. Ronneberger, P. Fischer and T. Brox (2015) U-net convolutional networks for biomedical image segmentation. *CoRR* **abs/2102.03881** https://arxiv.org/abs/2102.03881
73 M. Eliasof, T. Boesen, E. Haber, C. Keasar and E. Treister. 2021) Mimetic neural networks a unified framework for protein design and folding
74 J. Ingraham, V. Garg, R. Barzilay and T. Jaakkola (2019) Generative models for graph-based protein design. In: *Advances in Neural Information Processing Systems* (Wallach, H., Larochelle, H., Beygelzimer, A., d'Alch'e-Buc, F., Fox, E., Garnett. R., eds), vol. 32, Curran Associates, Inc., Red Hook, NY
75 X. Du, Y. Li, Y.L. Xia, S.M. Ai, J. Liang and P. Sang et al. (2016) Insights into protein–ligand interactions mechanisms, models, and methods. *Int. J. Mol. Sci.* **17**, 144 https://doi.org/10.3390/ijms17020144
76 T. Nguyen, H. Le and S. Venkatesh (2019) GraphDTA: prediction of drug–target binding affinity using graph convolutional networks. *bioRxiv*
77 H. Öztük, A. Özgür and E. Ozkirimli (2018) DeepDTA: deep drug–target binding affinity prediction. *Bioinformatics* **34**, i821–i829 https://doi.org/10.1093/bioinformatics/bty593
78 D. Antunes, D. Devaurs, M. Moll, G. Lizée and L. Kavraki (2018) General prediction of peptide-MHC binding modes using incremental docking a proof of concept. In *BCB '18: Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pp. 568–568, ACM https://doi.org/10.1145/3233547.3233719
79 D.A. Antunes, M. Moll, D. Devaurs, K.R. Jackson, G. Liz'ee and L.E. Kavraki (2017) DINC 2.0: a new protein–peptide docking webserver using an incremental approach. *Cancer Res.* **77**, e55–e57 https://doi.org/10.1158/0008-5472.CAN-17-0511
80 M.M. Rigo, D.A. Antunes, M.V. de Freitas, M.F. de Almeida Mendes, L. Meira and M. Sinigaglia et al. (2015) DockTope: a web-based tool for automated pMHC-I modelling. *Sci. Rep.* **5**, 18413 https://doi.org/10.1038/srep18413
81 D. Kozakov, D. Hall, B. Xia, K. Porter, D. Padhorny and C. Yueh et al. (2017) The ClusPro web server for protein-protein docking. *Nat. Protoc.* **12**, 255–278 https://doi.org/10.1038/nprot.2016.169
82 D. Kozakov, R. Brenke, S. Comeau and S. Vajda (2006) PIPER: an FFT-based protein docking program with pairwise potentials. *Proteins* **65**, 392–406 https://doi.org/10.1002/prot.21117
83 H. Ashtawy and N. Mahapatra (2015) Machine-learning scoring functions for identifying native poses of ligands docked to known and novel proteins. *BMC Bioinformatics* **16**, S3 https://doi.org/10.1186/1471-2105-16-S6-S3
84 H. Deng, Y. Jia and Y. Zhang (2017) Protein structure prediction. *Int. J. Modern Phys. B* **32**, 1840009 https://doi.org/10.1142/S021797921840009X
85 J. Cheng, M. Choe, A. Elofsson, K. Han, J. Hou and A. Maghrabi et al. (2019) Estimation of model accuracy in CASP13. *Proteins: Struct. Funct. Bioinformatics* **87**, 1361–1377 https://doi.org/10.1002/prot.v87.12
86 D. Lee, O. Redfern and C. Orengo (2008) Predicting protein function from sequence and structure. *Nat. Rev. Mol. Cell Biol.* **8**, 995–1005 https://doi.org/10.1038/nrm2281
87 M. Kulmanov, M.A. Khan and R. Hoehndorf (2017) DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics* **34**, 660–668 https://doi.org/10.1093/bioinformatics/btx624
88 H. Edelsbrunner and J. Harer (2008) Persistent homology: a survey. *Discrete Comput. Geom.* **453** https://doi.org/10.1090/conm/453/08802
89 I. Schomburg, A. Chang, C. Ebeling, M. Gremse, C. Heldt and G. Huhn et al. (2004) BRENDA, the enzyme database updates and major new developments. *Nucleic Acids Res.* **32** (Suppl 1), D431–D433 https://doi.org/10.1093/nar/gkh081
90 K.M. Borgwardt, C. Soon Ong, S. Schönauer, S.V.N. Vishwanathan, A. Smola and H.P. Kriegel et al. (2005) Protein function prediction via graph kernels. *Bioinformatics* **21** (Suppl 1), i47–i56 https://doi.org/10.1093/bioinformatics/bti1007

Emerging Topics in Life Sciences (2021) **5** 789–802
https://doi.org/10.1042/ETLS20210225

91 P. Dobson and A. Doig (2003) Distinguishing enzyme structures from non-enzymes without alignments. *J. Mol. Biol.* **330**, 771–783 https://doi.org/10.1016/S0022-2836(03)00628-4
92 V.P. Dwivedi, C.K. Joshi, T. Laurent, Y. Bengio and X. Bresson (2020) Benchmarking graph neural networks. *CoRR* **abs/2003.00982** https://arxiv.org/abs/2003.00982
93 J. Zhou, A.E. Panaitiu and G. Grigoryan (2020) A general-purpose protein design framework based on mining sequence–structure relationships in known protein structures. *Proc. Natl Acad. Sci. U.S.A.* **117**, 1059–1068 https://doi.org/10.1073/pnas.1908723117
94 J. Zhu, J. Wang, W. Han and D. Xu (2021) Neural relational inference to learn allosteric long-range interactions in proteins from molecular dynamics simulations. *bioRxiv*
95 T. Kipf, E. Fetaya, K.C. Wang, M. Welling and R. Zemel (2018) Neural relational inference for interacting systems, preprint, http://arxiv.org/180204687