Relational Pretraining for the Next Generation of Graph Intelligence

Alessandro Rossi
Sapienza University of Rome
alessandro.rossi@uniroma1.it

Francesca Bianchi
Polytechnic University of Milan
francesca.bianchi@polimi.it

Giancarlo Manuele
University of Pisa
giancarlo.manuele@unipi.it

Abstract

The rapid advancement of foundation models has transformed the landscape of machine learning by enabling scalable, general-purpose solutions across diverse domains such as natural language processing, computer vision, and multimodal reasoning. Simultaneously, Graph Neural Networks (GNNs) have become the de facto standard for learning over structured data due to their ability to model complex relational dependencies inherent in graphs. The convergence of these two paradigms—GNNs and foundation models—marks a significant milestone in the pursuit of universal representation learning over graph-structured data. This survey presents a comprehensive and in-depth exploration of Graph Neural Networks in the context of large-scale foundation models, highlighting key methodologies, architectural innovations, training strategies, applications, and open research challenges.

We begin by reviewing the mathematical underpinnings of graph neural architectures, including message passing, graph convolution, and attention mechanisms. Building on this foundation, we explore how self-supervised pretraining techniques—such as masked node and edge prediction, contrastive learning, and graph autoencoding—have been adapted to equip graph models with the capacity for transfer learning and generalization. We further analyze architectural trends that scale GNNs to foundation model capacities, including Graph Transformers, scalable neighborhood sampling methods, and structural encoding schemes. A detailed overview of applications illustrates the versatility of graph foundation models in real-world scenarios, including drug discovery, knowledge graph reasoning, recommender systems, scientific simulations, and fraud detection.

In addressing current limitations, we examine critical challenges such as computational scalability, data availability, heterogeneity, dynamic graph modeling, interpretability, and ethical considerations. We also identify key directions for future research, including the development of universal graph encoders, integration with other modalities, lifelong graph learning, and responsible AI deployment. By synthesizing recent advances and outlining a forward-looking roadmap, this survey aims to provide both a foundational reference and a strategic perspective for researchers, practitioners, and developers working at the intersection of graphs and large-scale machine learning. Ultimately, we argue that graph foundation models are poised to play a central role in the next generation of AI systems, enabling machines to reason over structured relational data with unprecedented depth, scale, and generality.

Keywords: Graph Neural Networks, Foundation Models, Graph Transformers, Pretraining, Self-supervised Learning, Structured Data, Representation Learning, Knowledge Graphs, Multi-modal Learning, Scalability

1 Introduction

In recent years, the convergence of Graph Neural Networks (GNNs) and large-scale foundation models has emerged as a transformative development in the landscape of machine learning and artificial intelligence. Graphs offer a natural and expressive way to represent structured data, capturing complex relationships between entities in domains ranging from social networks and biological systems to recommendation engines and scientific discovery. GNNs, designed to operate directly on such graph-structured data, have demonstrated remarkable success in tasks involving node classification, link prediction, graph classification, and beyond [1]. Concurrently, the advent of foundation models—large-scale, pretrained models capable of adapting to a multitude of downstream tasks—has reshaped the paradigm of learning across diverse modalities, including language, vision, and increasingly, structured and relational data [2]. Foundation models, exemplified by models such as BERT, GPT, and CLIP, are typically trained on massive corpora using self-supervised objectives, allowing them to learn rich, general-purpose representations. These models exhibit emergent capabilities, generalizability, and transferability, which make them particularly appealing for complex tasks with limited supervision. The potential synergy between GNNs and foundation models lies in leveraging the relational inductive bias of GNNs with the scalable learning and generalization capacities of large foundation models [3]. This hybridization promises to unlock new possibilities in modeling relational data at scale, enabling reasoning over knowledge graphs, enhancing molecular property prediction, pow-

ering large-scale recommender systems, and more. This survey aims to provide a comprehensive examination of the intersection between Graph Neural Networks and large foundation models. Specifically, we explore how foundational principles of GNNs are being adapted and scaled to fit within the paradigm of foundation models, as well as how pretraining strategies, architectural innovations, and application domains are evolving in this context. We begin by reviewing the fundamental components of GNNs, including message passing, graph convolution, and attention mechanisms, establishing a foundational understanding upon which large-scale extensions are built [4]. We then discuss the defining characteristics of foundation models, highlighting their training regimes, transfer capabilities, and architectural trends. Next, we delve into the core methodologies that integrate GNNs with foundation model principles. This includes pretraining techniques on large-scale graph data, such as masked node and edge prediction, contrastive learning, and graph autoencoding. We also examine multi-modal foundation models where graphs serve as one of several input modalities, such as in vision-language-graph transformers, and models where large language models (LLMs) are interfaced with graph-structured knowledge to enhance reasoning and retrieval [5]. A significant portion of this survey is dedicated to architectural advances that have been proposed to scale GNNs to the level of foundation models [6]. These include innovations in scalability (e.g., sampling-based training, graph sparsification, distributed training frameworks), parameter sharing and modular design, as well as adaptations of transformer architectures to graph data (Graph Transformers) [7]. We also highlight recent efforts in developing general-purpose graph encoders and universal graph representations, which mirror the objectives pursued in other modalities by foundation models [8]. In addition to methodological insights, we catalog prominent benchmark datasets and evaluation metrics that are instrumental in driving progress in this emerging field [9]. We analyze empirical results and provide comparative discussions to elucidate current capabilities and limitations of GNN-based foundation models. Moreover, we explore practical applications where this confluence is already making an impact, such as drug discovery, fraud detection, traffic prediction, and large-scale information retrieval [10]. Despite the promise of unifying GNNs and foundation models, several challenges remain. These include the scarcity of massive labeled graph datasets for pretraining, the heterogeneity and non-Euclidean nature of graph data, the computational demands of large-scale graph processing, and the difficulty in maintaining interpretability and trustworthiness in such complex systems. We provide an in-depth discussion of these open problems, along with a roadmap for future research directions that could catalyze further advances [11]. In summary, this survey articulates the growing body of research at the intersection of Graph Neural Networks and large foundation models, offering a detailed synthesis of current approaches, challenges,

and opportunities [12]. By illuminating the path forward, we hope to foster further innovation and collaboration in this dynamic and impactful area of machine learning [13].

2 Preliminaries and Background

In this section, we introduce the foundational concepts necessary to understand the interplay between Graph Neural Networks (GNNs) and large foundation models [14]. We begin by formalizing the graph structure and the notation used throughout this survey, followed by a detailed explanation of the GNN message-passing framework [15]. We then transition to a high-level overview of foundation models and their training objectives, setting the stage for a deeper exploration of their convergence with graph-based learning systems. Let a graph be denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes (or vertices) and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges representing pairwise relations between nodes. Each node $v \in \mathcal{V}$ is associated with a feature vector $\mathbf{x}_v \in \mathbb{R}^d$, and each edge $(u, v) \in \mathcal{E}$ may have an associated edge feature $\mathbf{e}_{uv} \in \mathbb{R}^k$. The adjacency matrix of the graph is denoted by $\mathbf{A} \in \{0, 1\}^{|\mathcal{V}| \times |\mathcal{V}|}$, where $\mathbf{A}_{uv} = 1$ if $(u, v) \in \mathcal{E}$ and $\mathbf{A}_{uv} = 0$ otherwise [16]. The canonical GNN model follows the message passing neural network (MPNN) paradigm, where node representations are iteratively updated by aggregating messages from their neighbors. At the l-th layer, the hidden representation of node v is denoted $\mathbf{h}_v^{(l)} \in \mathbb{R}^{d_l}$. The general update rule for message passing can be described as:

$$\mathbf{h}_v^{(l)} = \text{UPDATE}^{(l)} \left(\mathbf{h}_v^{(l-1)}, \text{AGGREGATE}^{(l)} \left(\left\{ \mathbf{m}_{uv}^{(l)} : u \in \mathcal{N}(v) \right\} \right) \right),$$

where $\mathcal{N}(v)$ denotes the set of neighbors of v, and $\mathbf{m}_{uv}^{(l)} = \text{MESSAGE}^{(l)} \left(\mathbf{h}_u^{(l-1)}, \mathbf{h}_v^{(l-1)}, \mathbf{e}_{uv} \right)$ is the message sent from node u to v. Common choices for the AGGREGATE function include sum, mean, or max operations, ensuring permutation invariance across neighborhoods [17]. The UPDATE function is typically a nonlinear transformation such as a multi-layer perceptron (MLP), gated recurrent unit (GRU), or a residual connection [18]. Several prominent GNN architectures instantiate this message-passing framework in specific ways [19]. Graph Convolutional Networks (GCNs) [20] simplify the message passing by assuming symmetric aggregation and linear transformations:

$$\mathbf{H}^{(l)} = \sigma \left(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l-1)} \mathbf{W}^{(l)} \right),$$

where $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix with self-loops, $\tilde{\mathbf{D}}$ is the corresponding degree matrix, $\mathbf{W}^{(l)}$ is a trainable weight matrix, and $\sigma(\cdot)$ is an activation function such as ReLU. In contrast, Graph Attention Networks (GATs) [21] introduce

learnable attention coefficients to weigh the importance of neighboring nodes:

$$\alpha_{uv}^{(l)} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top[\mathbf{W}\mathbf{h}_u^{(l-1)} \| \mathbf{W}\mathbf{h}_v^{(l-1)}]\right)\right)}{\sum_{k\in\mathcal{N}(v)} \exp\left(\text{LeakyReLU}\left(\mathbf{a}^\top[\mathbf{W}\mathbf{h}_k^{(l-1)} \| \mathbf{W}\mathbf{h}_v^{(l-1)}]\right)\right)},$$

where $\mathbf{a}$ and $\mathbf{W}$ are learnable parameters, and $\| \cdot \|$ denotes concatenation. On the other hand, foundation models are typically defined by their scale, generality, and adaptability [22]. Formally, a foundation model $\mathcal{F}_\theta$ is trained over a large-scale corpus $\mathcal{D} = \{x_i\}_{i=1}^N$ using a self-supervised loss $\mathcal{L}_{\text{pretrain}}$, with the objective of learning representations $\mathbf{z}_i = \mathcal{F}_\theta(x_i)$ that transfer well to a variety of downstream tasks. In the context of transformers, the core architecture is based on multi-head self-attention:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V},$$

where the query $\mathbf{Q}$, key $\mathbf{K}$, and value $\mathbf{V}$ matrices are learned projections of the input embeddings. Pretraining objectives include masked language modeling (MLM), next sentence prediction (NSP), contrastive learning, and reconstruction losses, all aimed at capturing general-purpose, semantic-rich representations. When applied to graphs, foundation model principles require substantial rethinking of input structure, objective formulation, and scalability [23]. Graphs are inherently non-Euclidean, lack a canonical ordering of nodes, and often vary in size, which violates many of the assumptions underpinning traditional transformer-based models. To mitigate these issues, extensions such as Graph-BERT, GROVER, and Graphormer have adapted transformer blocks to operate over graph structures by incorporating spatial encoding, relative positional biases, and novel attention masks that reflect graph topology [24]. Understanding these foundations enables us to explore the core ideas behind graph-based foundation models. The key challenge lies in devising pretraining strategies that exploit the structural regularities in graphs while maintaining the flexibility and generality of large models. The next sections delve into these strategies, architectural adaptations, and application-specific considerations in greater depth.

3 Pretraining Strategies for Graph Foundation Models

Pretraining lies at the heart of foundation models, serving as the primary means by which they acquire transferable knowledge before fine-tuning on downstream tasks. In the context of graph neural networks, pretraining strategies must account for the topological, relational, and often heterogeneous nature of graph-structured

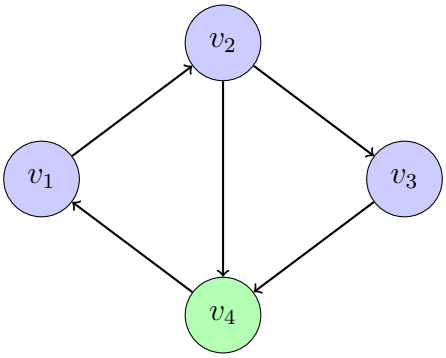

Figure 1: A simple directed graph with four nodes.

data [25]. Unlike textual or visual inputs, graphs lack a fixed sequence or grid structure, presenting unique challenges for developing effective self-supervised objectives. This section explores the dominant paradigms for graph pretraining, including node-level, edge-level, and graph-level objectives, as well as contrastive and generative learning approaches [26]. We also provide a summary comparison of these methods in Table 1. A widely adopted pretraining method in graph

Table 1: Summary of common graph pretraining strategies.

Method	**Objective Type**	**Granularity**	**Example Models**
Masked Node Prediction	Generative	Node-level	Graph-BERT, GROVER
Edge Prediction	Generative	Edge-level	PretrainGNN, CoAtGIN
Contextual Subgraph Prediction	Generative	Subgraph-level	InfoGraph, GraphMVP
Contrastive Learning	Discriminative	Node/Graph-level	DGI, GraphCL, BGRL
Graph Autoencoding	Reconstruction	Graph-level	GAE, VGAE

learning is masked node prediction, which draws direct inspiration from masked language modeling (MLM) in NLP. In this approach, a subset of node features $\{\mathbf{x}_v\}_{v \in \mathcal{M}}$ is masked out during training, and the model is tasked with reconstructing the original features based on the surrounding graph context. This encourages the model to capture both local and global structural dependencies [27]. Unlike textual data where word order provides strong contextual cues, the challenge in

graphs lies in learning to reason over neighborhoods defined by topological, rather than sequential, proximity. Edge prediction pretraining strategies similarly adopt a generative stance. Given two nodes u and v, the model is trained to predict whether an edge $(u, v) \in \mathcal{E}$ exists [28]. This binary classification task can be extended to model edge types or weights in heterogeneous or weighted graphs. Edge prediction has been particularly useful in molecular graphs, where chemical bond prediction serves as an effective pretraining signal [29]. More advanced variants include predicting entire neighborhoods or edge substructures, leading to richer representations. Subgraph-level pretraining introduces a hierarchical component to representation learning [30]. Rather than operating on single nodes or edges, models are trained to predict properties or embeddings of sampled subgraphs. For instance, contrastive subgraph learning methods generate multiple views of a graph—via node dropping, edge perturbation, or attribute masking—and enforce representation similarity across these views using contrastive loss functions such as InfoNCE [31]. These objectives do not rely on explicit labels, making them highly scalable and domain-agnostic [32]. Graph contrastive learning (GCL) has gained significant attention due to its robustness and empirical effectiveness. Models such as Deep Graph Infomax (DGI) and GraphCL formulate pretraining as a task of distinguishing positive pairs (e.g., perturbed versions of the same graph or node) from negative samples drawn from different graphs. This discriminative formulation encourages invariance to perturbations while preserving structural identity. More recent advancements include bootstrapped contrastive methods, such as BGRL and SimGRACE, which avoid negative sampling and optimize consistency across model augmentations. Generative graph autoencoders form another important category, where the encoder learns latent representations that are subsequently decoded to reconstruct either node features or adjacency structures. Variants such as variational graph autoencoders (VGAE) provide a probabilistic interpretation, learning distributions over latent graph embeddings. These methods have been effective in unsupervised node clustering, anomaly detection, and link prediction tasks, and have been scaled to larger graphs using sampling techniques [33]. Overall, the choice of pretraining objective in graph foundation models depends on several factors, including the graph domain, availability of auxiliary data, computational budget, and target downstream tasks. While generative approaches promote semantic richness, contrastive methods emphasize robustness and invariance. Hybrid techniques that combine generative and contrastive elements are an emerging trend, aiming to capture the strengths of both paradigms [34]. Moreover, domain-specific objectives—such as 3D conformer prediction in molecular graphs or reasoning over entity types in knowledge graphs—are increasingly being integrated into general-purpose pretraining schemes. As research progresses, a key question is how to design pretraining strategies that generalize across differ-

ent types of graphs (e.g., homogeneous, heterogeneous, attributed, dynamic) while maintaining the scalability and modularity of foundation models [35]. Future work will likely explore task-conditioned pretraining, meta-learning approaches, and co-training with large language or vision models to exploit multi-modal signals in graphs.

4 Architectural Innovations in Scalable Graph Foundation Models

As Graph Neural Networks (GNNs) evolve to meet the scale and generality demanded by foundation models, architectural innovation becomes a central pillar of progress [36]. Traditional GNNs, though powerful on small to medium-sized graphs, struggle with scalability, depth, and expressivity when applied to large heterogeneous graphs or multi-relational knowledge bases. To address these limitations, researchers have introduced a range of architectural modifications inspired by both transformer models and principles unique to graph structures. This section surveys key architectural advances, including Graph Transformers, scalable GNN backbones, efficient neighborhood aggregation schemes, and structural encoding strategies [37]. To aid intuition, Figure 2 provides a simplified illustration of a typical graph transformer architecture [38].

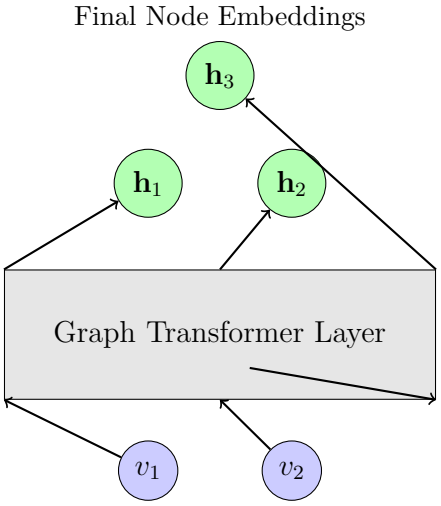

Figure 2: A simplified schematic of a Graph Transformer architecture. Raw node features are passed into a transformer-style encoder, which processes neighborhood context and outputs contextualized node embeddings.

One of the most significant developments in recent years has been the adapta-

tion of the transformer architecture to graphs. While transformers are inherently permutation-invariant and capable of modeling long-range dependencies, they require modification to respect graph topology. Early efforts such as Graph-BERT introduced spatial positional encodings, derived from shortest-path distances, to guide the attention mechanism. More recent models, including Graphormer and SAN-GNN, embed structural bias directly into the self-attention mechanism, allowing the model to weigh nodes based on both feature similarity and topological distance. A typical Graph Transformer replaces the adjacency-based message passing in conventional GNNs with full attention across all node pairs, modulated by graph-aware positional encodings. Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the attention weight between two nodes u and v is defined as:

$$\alpha_{uv} = \frac{\exp\left(\frac{(\mathbf{Q}_u + \mathbf{p}_u)^\top (\mathbf{K}_v + \mathbf{p}_v)}{\sqrt{d}}\right)}{\sum_{k \in \mathcal{V}} \exp\left(\frac{(\mathbf{Q}_u + \mathbf{p}_u)^\top (\mathbf{K}_k + \mathbf{p}_k)}{\sqrt{d}}\right)},$$

where $\mathbf{Q}_u$, $\mathbf{K}_v$ are the query and key projections of nodes u and v, and $\mathbf{p}_u$, $\mathbf{p}_v$ are position encodings derived from graph distances, centrality, or Laplacian eigenvectors [39]. These architectural choices allow the model to capture global context without discarding local graph structure. Scalability, however, remains a critical concern, especially for massive graphs with millions or billions of nodes. Full-attention mechanisms have quadratic time and memory complexity with respect to the number of nodes. To address this, models such as Cluster-GCN, GraphSAINT, and LADIES employ sampling strategies to construct mini-batches that preserve neighborhood structure while limiting computational cost. Other methods, like GShard-based distributed training or sparse attention mechanisms, push scalability further by partitioning computation across clusters and minimizing redundant communication [40]. In addition to attention-based innovations, modular and hierarchical designs have proven useful in scaling graph models. For instance, scalable architectures often include bottleneck layers, global readouts, or virtual nodes to facilitate long-range information flow. Techniques like Jumping Knowledge (JK) networks and residual connections alleviate the problem of over-smoothing in deep GNNs, enabling deeper architectures that maintain node-level expressivity [41]. Furthermore, a crucial architectural consideration in graph foundation models is the encoding of structural inductive biases [42]. Since graphs do not possess a natural ordering or spatial layout, these biases must be injected explicitly. This includes using structural features such as degree, PageRank, betweenness centrality, or graphlets as input augmentations [43]. Positional encoding strategies based on Laplacian eigenvectors, persistent homology, or random walk distributions provide rich signals for modeling graph topology in a permutation-invariant way. As a final note, hybrid architectures that blend GNN modules with transformer blocks are gaining popularity. These designs use GNN layers to pro-

cess local neighborhoods efficiently and transformer layers to incorporate global context. The resulting systems combine the best of both worlds: the relational inductive bias of GNNs and the generalization power of attention-based models [44]. Such hybrids are particularly well-suited for large-scale, multi-domain graph tasks such as molecular property prediction, knowledge graph reasoning, and scientific discovery. Overall, the landscape of graph model architectures is rapidly evolving toward greater scalability, flexibility, and universality [45]. Future research is likely to explore dynamic graph architectures, efficient training on streaming or temporal graphs, and tighter integration with multi-modal foundation models, where graph modules coexist with language and vision components under unified learning frameworks [46].

5 Applications and Use Cases of Graph Foundation Models

The integration of Graph Neural Networks with the scale and adaptability of foundation models has significantly broadened the landscape of real-world applications that can benefit from graph-based reasoning and representation learning. Unlike conventional GNNs that are often fine-tuned for specific datasets or tasks, graph foundation models aim to learn general-purpose representations that are transferable across domains, reducing the reliance on task-specific engineering and labeled data. This section explores some of the most impactful application domains where these models have demonstrated significant promise, including drug discovery, knowledge graph completion, recommendation systems, scientific discovery, and financial fraud detection. One of the earliest and most well-aligned domains for graph foundation models is computational chemistry, particularly drug discovery and molecular property prediction [47]. Molecules are naturally represented as graphs, with atoms as nodes and chemical bonds as edges [48]. Pretraining large graph encoders on vast corpora of unlabeled molecular structures—such as those in the ZINC, ChEMBL, and PubChem databases—allows models to learn representations that encapsulate chemical reactivity, geometric configurations, and substructure motifs. Models such as GROVER and GraphMVP have achieved state-of-the-art results on downstream tasks like toxicity prediction, solubility estimation, and bioactivity classification [49]. Moreover, when integrated with 3D-aware modules and quantum-level simulations, these models enable powerful zero-shot generalization to new compounds, accelerating the discovery of novel drugs, materials, and catalysts [50]. Another domain that has been transformed by graph-based foundation models is knowledge graph completion and reasoning. Knowledge graphs, such as Freebase, DBpedia, and ConceptNet, encode facts in

the form of triplets (h, r, t), representing head entities, relations, and tail entities [51]. Graph foundation models pretrained on large-scale multi-relational graphs have demonstrated the ability to generalize to unseen entities and relations via transfer learning. These models enable not only link prediction (e.g., inferring missing facts) but also logical reasoning, path-based inference, and question answering when integrated with large language models (LLMs). Recent efforts in neuro-symbolic reasoning combine symbolic logic over graphs with the semantic richness of deep embeddings, bridging the gap between statistical and symbolic AI. In the realm of recommender systems, user-item interactions can naturally be modeled as bipartite graphs. By pretraining graph encoders on large-scale interaction data—such as clicks, ratings, purchases, or browsing histories—foundation models can generate rich embeddings that capture both user preferences and item semantics. These representations are robust to sparsity, temporal drift, and cold-start scenarios. Techniques such as contrastive graph pretraining and session-based augmentation enhance the model's ability to recommend items that are novel, diverse, and personalized [52]. Some commercial systems now integrate graph pretraining with cross-modal encoders (e.g., text and image representations) to deliver unified recommendation pipelines that operate at web scale. Scientific discovery is another fertile ground for graph foundation models, especially in physics, materials science, and biology, where structured data is abundant but labeled data is scarce [53]. For example, in protein-protein interaction networks, nodes represent proteins and edges represent functional or physical associations. Pretrained graph models can predict protein functions, infer missing interactions, and classify diseases based on interaction patterns. In physics, graph neural operators have been used to simulate complex systems like fluid dynamics and particle interactions by learning continuous operators over graph-structured state spaces [54]. When these models are scaled and pretrained across datasets with varying physical laws, they exhibit a form of scientific transfer learning—generalizing knowledge across tasks that obey different underlying dynamics [55]. In finance and cybersecurity, graphs are instrumental in modeling complex transaction networks, fraud rings, and malware propagation. Nodes may represent entities such as users, devices, or bank accounts, and edges encode financial transactions or communication events [56]. Foundation models pretrained on massive transactional graphs can detect anomalous patterns, identify collusive behaviors, and uncover fraudulent subgraphs, even when adversarial behavior evolves dynamically. The ability of these models to generalize from historical patterns and adapt to novel attack strategies makes them particularly valuable in adversarial and high-stakes environments. Despite these advances, deploying graph foundation models in practice presents a number of challenges [57]. These include data privacy concerns, the interpretability of learned representations, domain adaptation issues, and the difficulty of aligning pretrained

models with the specific constraints of operational systems [58]. Additionally, fine-tuning large-scale graph models remains computationally expensive, particularly in dynamic settings where the underlying graph structure is constantly evolving [59]. Nonetheless, the promise of graph foundation models to offer scalable, robust, and adaptable solutions across a spectrum of applications is becoming increasingly tangible [60]. As tooling, datasets, and compute infrastructures continue to mature, we expect to see widespread adoption of these models across industries and research domains.

6 Challenges and Open Problems

While the development of graph foundation models represents a significant leap forward in the ability to process, reason over, and generalize from structured relational data, numerous challenges remain that hinder their full realization and deployment [61]. These challenges are both theoretical and practical in nature, spanning issues of scalability, heterogeneity, data sparsity, interpretability, and ethical deployment. Addressing these challenges is critical to enabling robust, generalizable, and trustworthy graph-based models that can operate reliably in real-world settings [62]. A primary challenge lies in the scalability of graph foundation models. Unlike text or images, which can be processed in mini-batches with consistent structure, graphs are inherently irregular, often massive, and interconnected in a way that makes standard batching inefficient. The computational complexity of full-graph transformers grows quadratically with the number of nodes, rendering naïve application of attention mechanisms infeasible for large graphs such as social networks or web-scale knowledge graphs [63]. Although approximation methods (e.g., sampling, subgraph training, or sparse attention) offer partial solutions, they often involve trade-offs between accuracy and efficiency[64, 65]. Moreover, such techniques may obscure global patterns or long-range dependencies that are critical for some tasks. Future work must continue to explore architectures and training algorithms that are both computationally scalable and information-rich, such as locality-sensitive hashing, graph coarsening, or distributed GNN frameworks with communication-aware designs [66]. Another major open problem is the lack of high-quality, large-scale, and diverse graph datasets suitable for pretraining. In natural language processing, foundation models benefit from access to trillions of tokens from open-domain corpora. By contrast, the availability of graph data is fragmented, often domain-specific, and sometimes encumbered by privacy and ownership concerns. Many publicly available graph datasets (e.g., Cora, Citeseer, Pubmed) are small and saturated in terms of benchmarking utility. Even larger datasets such as ogbn-products or ogbl-collab are limited in diversity and do not reflect the heterogeneous nature of many real-world graphs. There is an urgent need

for standardized, representative graph pretraining corpora that encompass various types of structures—social, biological, temporal, multi-relational—and that support cross-domain transfer learning. Heterogeneity and dynamics further complicate the design of universal graph foundation models. Real-world graphs are rarely homogeneous; they may involve multiple node and edge types, attributes with varying semantics, and temporal evolution over time. Capturing this heterogeneity in a unified representation space requires sophisticated modeling of type-specific transformations, temporal dependencies, and contextual interactions [67]. Furthermore, dynamic graphs—where nodes and edges appear, disappear, or change over time—pose additional challenges for both pretraining and inference [68]. Current GNNs and graph transformers often assume static input, limiting their applicability in streaming or time-sensitive environments such as financial transactions or communication networks. Extensions to dynamic, temporal-aware graph models that maintain consistency across time and enable continual learning are a crucial direction for research. Interpretability and explainability remain another persistent bottleneck [69]. While GNNs offer some level of interpretability via attention weights or neighborhood aggregation patterns, the complexity of deep, pretrained graph models makes it difficult to attribute predictions to specific parts of the graph. This issue is exacerbated in safety-critical domains such as healthcare or law enforcement, where trust and accountability are essential [70]. Although post-hoc explanation methods (e.g., GNNExplainer, PGExplainer) exist, they are often heuristic and lack theoretical guarantees. More robust, intrinsic approaches to interpretability, possibly drawing from causal inference or symbolic reasoning, are needed to ensure that graph foundation models are not just powerful, but also transparent and trustworthy [71]. In addition to technical challenges, ethical and societal considerations must not be overlooked. Graph data is often rich in sensitive information, particularly when modeling human behavior, relationships, or financial transactions. The risk of amplifying biases, leaking private information, or enabling malicious inference increases with the scale and generality of pretrained models. Furthermore, foundation models, by virtue of being pretrained on large corpora, can inadvertently encode and propagate harmful stereotypes or structural inequalities [72]. Responsible design and deployment of graph foundation models therefore requires careful attention to fairness, accountability, and interpretability. Techniques such as differential privacy for graph data, fairness-aware training objectives, and federated graph learning offer promising directions to mitigate these risks [73]. Finally, there remains a fundamental question regarding the universality of graph foundation models: to what extent can a single pretrained model generalize across diverse graph domains [74]? While multimodal foundation models in NLP and vision have demonstrated remarkable transferability, graphs are less amenable to unified representation due to the diversity of their topology,

semantics, and size. Understanding the limits of transfer in graph models, and developing meta-learning or domain adaptation techniques to enhance cross-domain generalization, is an important area of future research. In summary, while graph foundation models hold immense potential, their full realization is contingent upon overcoming a complex array of theoretical, computational, and ethical challenges. Addressing these open problems will not only enhance the performance and applicability of graph-based AI systems but also ensure their responsible and equitable deployment across society.

7 Future Directions

As graph foundation models continue to evolve, several compelling avenues of research and development are emerging that will define the next generation of structured AI systems. These future directions span improvements in scalability and generalization, integration with other modalities, deeper theoretical understanding, and advancements in training efficiency, robustness, and democratized access. Together, these themes represent a multi-dimensional roadmap for the ongoing progress in this field [75]. One of the most promising directions is the design of **truly universal graph encoders** that can operate across a broad spectrum of graph types—homogeneous, heterogeneous, dynamic, and multi-relational—without requiring architectural customization for each domain. Current models often rely on fixed assumptions about the graph structure, such as undirected edges or static topology, which limits their transferability. Future graph foundation models must generalize over variations in node and edge types, support evolving graph structures, and dynamically adjust their message-passing strategies depending on context. Meta-learning techniques, adaptive inductive biases, and task-conditioned encoding layers are likely to play a central role in building such flexible architectures [76]. This would allow a single pretrained model to adapt efficiently to new graph domains with minimal supervision, mirroring the success seen in large language models across diverse linguistic tasks [77]. A second area of opportunity is the **integration of graphs with other data modalities**—including text, images, time series, and tabular data—to enable joint reasoning over multi-modal inputs. Graphs often serve as the underlying structure that links heterogeneous data sources, such as connecting documents in citation networks, images in scene graphs, or proteins in biological interaction networks [78]. Future foundation models should treat graph information as a first-class citizen within a broader multi-modal learning framework, where node features could be text embeddings, edge features could encode image similarity, and global graph context could be captured through temporal or spatial cues. Developing joint architectures that harmonize graph transformers with vision and language

backbones, possibly through shared attention mechanisms or modality-specific encoders with cross-attention bridges, is an exciting and largely untapped research frontier [79]. Another fundamental direction is the development of **theoretical frameworks to understand the expressivity and limitations of graph foundation models** [80]. While empirical progress has been substantial, there is a lack of rigorous theory to explain why certain architectural choices work well across tasks and datasets. Concepts such as graph isomorphism tests, Weisfeiler-Lehman hierarchy, and spectral graph theory have helped to classify GNNs, but analogous tools are lacking for large pretrained models. New theoretical insights could elucidate the trade-offs between local and global representation power, the effects of over-smoothing in deep layers, or the dynamics of attention-based message passing on sparse topologies [81]. Such theory is essential not only for guiding architecture design but also for ensuring reliability and robustness in safety-critical deployments [82]. Moreover, **training efficiency and resource accessibility** remain pressing concerns. Current graph foundation models are extremely data- and compute-intensive to train, often requiring specialized infrastructure and access to proprietary datasets [83]. To democratize this technology, there is a growing need for efficient training techniques such as knowledge distillation, low-rank adaptation (LoRA), parameter-efficient fine-tuning (PEFT), and sparsity-aware computation. Furthermore, advances in curriculum learning, self-supervised data filtering, and progressive model scaling could allow smaller organizations and academic researchers to train or adapt powerful graph models without incurring prohibitive costs [84]. These methods must also be accompanied by open-source tools, pretrained checkpoints, and reproducible benchmarks that lower the barrier to entry for the broader community. Another vital future direction is the development of **continual and lifelong graph learning paradigms**, where models can be incrementally updated as new data arrives, without retraining from scratch or forgetting previous knowledge [85]. In real-world applications—such as recommendation systems, fraud detection, or scientific databases—graphs are constantly evolving, and static pretrained models quickly become outdated. Techniques for continual graph learning must balance stability and plasticity, enabling adaptation to new nodes, edges, and patterns while retaining historical accuracy. This will involve innovations in memory-augmented architectures, dynamic positional encoding, and online learning algorithms that respect the structural constraints of graphs. Finally, the **ethical and responsible deployment** of graph foundation models is a crucial area that must evolve in parallel with technical progress [86]. These models increasingly inform decisions in high-stakes domains, from finance to medicine to governance [87]. As such, mechanisms for ensuring fairness, detecting bias, safeguarding privacy, and providing interpretable outputs are non-negotiable. There is a need for formal frameworks that can quantify and mitigate risks associated

with graph representation learning, particularly in settings where graph structure itself can encode or amplify societal biases (e.g., in social or criminal networks) [88]. Fairness-aware pretraining, adversarial auditing of graph embeddings, and compliance-aware design principles will be essential for aligning graph foundation models with broader human values [89]. In conclusion, the future of graph foundation models is rich with possibility and challenge. By advancing architecture, theory, training, and application simultaneously—and by keeping ethical and societal impacts front and center—researchers and practitioners can build graph-based AI systems that are not only powerful and generalizable but also trustworthy, inclusive, and beneficial to the global community.

8 Conclusion

The fusion of Graph Neural Networks and large-scale foundation models represents a pivotal evolution in the trajectory of artificial intelligence. As structured data becomes increasingly prevalent in domains such as biology, chemistry, finance, and the web, the ability to reason over relational information at scale is no longer a niche requirement—it is foundational. Graph foundation models seek to meet this need by combining the inductive biases of graph-based representation learning with the scalability, generalization, and adaptability of modern pretrained models. This synthesis has already begun to reshape a wide variety of application domains, from molecular property prediction to knowledge reasoning and recommender systems, offering significant improvements in performance, robustness, and versatility.

Throughout this survey, we have explored the foundations, methodologies, architectural advances, and applications of graph foundation models. We began with a detailed review of the underlying mathematical formulations of graph neural networks and their extension into message-passing frameworks. Building on this, we examined how pretraining strategies—whether generative, contrastive, or hybrid—equip graph encoders with transferable knowledge, enabling strong downstream performance even in low-label regimes. We then delved into the architectural innovations that facilitate the scaling of GNNs to the level of foundation models, including Graph Transformers, attention-based encoding, and sampling techniques that preserve global context while maintaining computational efficiency.

Applications across science, industry, and social systems demonstrate the real-world value of graph foundation models. These models are capable not only of delivering state-of-the-art results on benchmark tasks, but also of supporting tasks that are structurally complex and previously out of reach for traditional deep learning architectures. In drug discovery, for instance, pretrained graph models accelerate the exploration of vast chemical spaces; in knowledge graphs, they enable sophisticated reasoning over incomplete and noisy data. As these models are

increasingly deployed, they provide a foundation for more intelligent, structure-aware systems that can learn, reason, and adapt in real time.

Despite these advances, several formidable challenges remain. Issues of scalability, interpretability, and fairness are far from solved. The irregular, non-Euclidean nature of graph data complicates standard training and evaluation pipelines, while the lack of large, diverse graph corpora hinders the development of truly general-purpose graph foundation models. Moreover, concerns about bias, privacy, and responsible use are magnified by the structural sensitivity of graphs, where small changes in connectivity can lead to large shifts in learned representations or predictions. Meeting these challenges requires not only technical innovation, but also interdisciplinary collaboration that brings together machine learning, graph theory, ethics, and domain-specific knowledge.

Looking forward, the future of graph foundation models will likely be defined by unification and integration. Models will become more universal—capable of handling arbitrary graph types and operating across diverse domains with minimal adaptation. They will also become more multimodal, combining graph structure with language, vision, and other sensory inputs to support richer forms of perception and reasoning. Theoretical progress will provide the foundations for understanding and improving these systems, while practical tools will make them accessible to a broader community of developers, researchers, and decision-makers.

In conclusion, graph foundation models represent a major milestone on the path toward general-purpose, structure-aware artificial intelligence. They extend the reach of deep learning beyond sequences and grids into the relational fabric of real-world data, enabling machines to learn from and interact with complex systems in ways that are both powerful and principled. By continuing to advance this field—through better architectures, smarter training strategies, and more ethical deployment practices—we can build AI systems that are not only technically capable but also aligned with the needs and values of the societies they serve.

References

[1] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.

[2] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. Representation learning on graphs with jumping knowledge networks. In *International conference on machine learning*, pages 5453–5462. PMLR, 2018.

[3] Yangze Zhou, Gitta Kutyniok, and Bruno Ribeiro. OOD link prediction generalization capabilities of message-passing GNNs in larger test graphs. *NeurIPS*, 2022.

[4] Yixuan He, Xitong Zhang, Junjie Huang, Benedek Rozemberczki, Mihai Cucuringu, and Gesine Reinert. Pytorch geometric signed directed: a software package on graph neural networks for signed and directed graphs. In *Learning on Graphs Conference*, pages 12–1. PMLR, 2024.

[5] Shuhan Liu and Kaize Ding. Beyond generalization: A survey of out-of-distribution adaptation on graphs. *CoRR*, abs/2402.11153, 2024.

[6] Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. *arXiv preprint arXiv:2012.09699*, 2020.

[7] L. Babai. Graph isomorphism in quasipolynomial time. In *Symposium on Theory of Computing*, pages 684–697, 2016.

[8] Wengong Jin, Connor W. Coley, Regina Barzilay, and Tommi S. Jaakkola. Predicting organic reaction outcomes with weisfeiler-lehman network. In *NeurIPS 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 2607–2616, 2017.

[9] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. Protein interface prediction using graph convolutional networks. In *NeurIPS 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6530–6539, 2017.

[10] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. Cord: a consolidated receipt dataset for post-ocr parsing. In *Workshop on Document Intelligence at NeurIPS 2019*, 2019.

[11] Antonio Longa, Veronica Lachi, Gabriele Santin, Monica Bianchini, Bruno Lepri, Pietro Lio, Franco Scarselli, and Andrea Passerini. Graph neural networks for temporal graphs: State of the art, open challenges, and opportunities. *arXiv preprint arXiv:2302.01018*, 2023.

[12] Meng Liu, Hongyang Gao, and Shuiwang Ji. Towards deeper graph neural networks. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 338–348, 2020.

[13] Monica Bianchini and Franco Scarselli. On the complexity of neural network classifiers: A comparison between shallow and deep architectures. *IEEE*

Transactions on Neural Networks and Learning Systems, 25(8):1553–1565, 2014.

[14] Benjamin Chamberlain, James Rowbottom, Davide Eynard, Francesco Di Giovanni, Xiaowen Dong, and Michael Bronstein. Beltrami flow and neural diffusion on graphs. *Advances in Neural Information Processing Systems*, 34: 1594–1609, 2021.

[15] Elias B. Khalil, Hanjun Dai, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs. In *NeurIPS 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6348–6358, 2017.

[16] Mahito Sugiyama and Karsten M. Borgwardt. Halting in random walk kernels. In *NeurIPS 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1639–1647, 2015.

[17] Álvaro Arroyo, Alessio Gravina, Benjamin Gutteridge, Federico Barbero, Claudio Gallicchio, Xiaowen Dong, Michael Bronstein, and Pierre Vandergheynst. On vanishing gradients, over-smoothing, and over-squashing in gnns: Bridging recurrent and graph learning. *arXiv preprint arXiv:2502.10818*, 2025.

[18] Zaïd Harchaoui and Francis R. Bach. Image classification with segmentation graph kernels. In *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), 18-23 June 2007, Minneapolis, Minnesota, USA*, 2007.

[19] Joan Andreu Sánchez, Verónica Romero, Alejandro H Toselli, Mauricio Villegas, and Enrique Vidal. A set of benchmarks for handwritten text recognition on historical documents. *Pattern Recognition*, 94:122–134, 2019.

[20] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[21] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[22] Hongwei Wang, Jia Wang, Jialin Wang, Miao Zhao, Weinan Zhang, Fuzheng Zhang, Xing Xie, and Minyi Guo. Graphgan: Graph representation learning with generative adversarial nets. In *Proc. of AAAI*, pages 2508–2515, 2018.

[23] Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 922–929, 2019.

[24] Yuliang Liu, Zhang Li, Biao Yang, Chunyuan Li, Xucheng Yin, Cheng-lin Liu, Lianwen Jin, and Xiang Bai. On the hidden mystery of ocr in large multimodal models. *arXiv e-prints*, pages arXiv–2305, 2023.

[25] O. Goldreich. Introduction to testing graph properties. In *Property Testing*. Springer, 2010.

[26] F. D. Johansson, O. Frost, C. Retzner, and D. Dubhashi. Classifying large graphs with differential privacy. In *Modeling Decisions for Artificial Intelligence*, pages 3–17, 2015.

[27] S. Hido and H. Kashima. A linear-time graph kernel. In *IEEE International Conference on Data Mining*, pages 179–188, 2009.

[28] Francesco Di Giovanni. Rotationally symmetric ricci flow on rn+ 1. *Advances in Mathematics*, 381:107621, 2021.

[29] B. Weisfeiler. *On Construction and Identification of Graphs*. Springer, 1976.

[30] Vladimir Vapnik. *Estimation of Dependences Based on Empirical Data, Second Editiontion*. Springer, 2006.

[31] Lorenzo Giusti, Claudio Battiloro, Lucia Testa, Paolo Di Lorenzo, Stefania Sardellitti, and Sergio Barbarossa. Cell attention networks. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2023.

[32] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. Trocr: Transformer-based optical character recognition with pre-trained models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 13094–13102, 2023.

[33] J. Stokes, K. Yang, K. Swanson, W. Jin, A. Cubillos-Ruiz, N. Donghia, C. MacNair, S. French, L. Carfrae, Z. Bloom-Ackerman, V. Tran, A. Chiappino-Pepe, A. Badran, I. Andrews, E. Chory, G. Church, E. Brown, T. Jaakkola, R. Barzilay, and J. Collins. A deep learning approach to antibiotic discovery. *Cell*, pages 688–702.e13, 2020.

[34] M. Wang, L. Yu, D. Zheng, Q. Gan, Y. Gai, Z. Ye, M. Li, J. Zhou, Q. Huang, C. Ma, Z. Huang, Q. Guo, H. Zhang, H. Lin, J. Zhao, J. Li, A. J. Smola, and Z. Zhang. Deep graph library: Towards efficient and scalable deep learning on graphs. *arXiv preprint*, 2019.

[35] Ruslan Salakhutdinov. Deep learning. In *Proc. of KDD*, page 1973, 2014.

[36] S. Kiefer, I. Ponomarenko, and P. Schweitzer. The Weisfeiler-Leman dimension of planar graphs is at most 3. *Journal of the ACM*, 66(6):44:1–44:31, 2019.

[37] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*, 2023.

[38] D. B. Kireev. Chemnet: A novel neural network based method for graph/property mapping. *Journal of Chemical Information and Computer Sciences*, 35(2):175–180, 1995.

[39] P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *ACM Symposium on Theory of Computing*, pages 604–613, 1998.

[40] J. Wang, S. Zhang, Y. Xiao, and R. Song. A review on graph neural network methods in financial applications. *arXiv preprint*, arXiv:2111.15367, 2021.

[41] Chen Cai, Truong Son Hy, Rose Yu, and Yusu Wang. On the connection between mpnn and graph transformer. In *ICML*, 2023.

[42] Zeyu Zhang, Jiamou Liu, Xianda Zheng, Yifei Wang, Pengqian Han, Yupan Wang, Kaiqi Zhao, and Zijian Zhang. Rsgnn: A model-agnostic approach for enhancing the robustness of signed graph neural networks. In *Proceedings of the ACM Web Conference 2023*, pages 60–70, 2023.

[43] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *NeurIPS*, 2020.

[44] Pál András Papp, Karolis Martinkus, Lukas Faber, and Roger Wattenhofer. Dropgnn: Random dropouts increase the expressiveness of graph neural networks. *Advances in Neural Information Processing Systems*, 34:21997–22009, 2021.

[45] Yibo Yao and Lawrence B. Holder. Scalable classification for large dynamic networks. In *2015 IEEE International Conference on Big Data, Big Data 2015, Santa Clara, CA, USA, October 29 - November 1, 2015*, pages 609–618, 2015.

[46] Eli Meirom, Haggai Maron, Shie Mannor, and Gal Chechik. Optimizing tensor network contraction using reinforcement learning. In *ICLR*. PMLR, 2022.

[47] Matteo Riondato and Eli Upfal. ABRA: approximating betweenness centrality in static and dynamic graphs with rademacher averages. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1145–1154, 2016.

[48] Xianchen Zhou and Hongxia Wang. The generalization error of graph convolutional networks may enlarge with more layers. *Neurocomputing*, 424, 2021.

[49] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

[50] Y. Huang, X. Peng, J. Ma, and M. Zhang. Boosting the cycle counting power of graph neural networks with I^2-GNNs. *arXiv preprint*, 2022.

[51] Franco Manessi, Alessandro Rozza, and Mario Manzo. Dynamic graph convolutional networks. *Pattern Recognition*, 97:107000, 2020.

[52] Guangyong Chen, Pengfei Chen, Chang-Yu Hsieh, Chee-Kong Lee, Benben Liao, Renjie Liao, Weiwen Liu, Jiezhong Qiu, Qiming Sun, Jie Tang, Richard S. Zemel, and Shengyu Zhang. Alchemy: A quantum chemistry dataset for benchmarking AI models. *arXiv preprint*, 2019.

[53] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured sequence modeling with graph convolutional recurrent networks. arxiv (2016). *arXiv preprint arXiv:1612.07659*, 2016.

[54] Ralph Abboud, Radoslav Dimitrov, and Ismail Ilkan Ceylan. Shortest path networks for graph property prediction. In *Learning on Graphs Conference*, pages 5–1. PMLR, 2022.

[55] Francesco Orsini, Paolo Frasconi, and Luc De Raedt. Graph invariant kernels. In *Proceedings of the Twenty-Fourth IJCAI, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 3756–3762, 2015.

[56] Duncan J Watts and Steven H Strogatz. Collective dynamics of 'small-world'networks. *Nature*, (6684):440, 1998.

[57] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 5308–5317, 2016.

[58] M. Simonovsky and N. Komodakis. GraphVAE: Towards generation of small graphs using variational autoencoders. In *Artificial Neural Networks and Machine Learning*, pages 412–422, 2018.

[59] Luana Ruiz, Luiz F. O. Chamon, and Alejandro Ribeiro. Graphon neural networks and the transferability of graph neural networks. In *NeurIPS*, 2020.

[60] J. Ramon and T. Gärtner. Expressivity versus efficiency of graph kernels. In *International Workshop on Mining Graphs, Trees and Sequences*, 2003.

[61] Federico Barbero, Ameya Velingker, Amin Saberi, Michael Bronstein, and Francesco Di Giovanni. Locality-aware graph-rewiring in gnns. *arXiv:2310.01668*, 2023.

[62] Jacob Bamberger, Federico Barbero, Xiaowen Dong, and Michael M. Bronstein. Bundle neural network for message diffusion on graphs. In *The Thirteenth International Conference on Learning Representations*, 2025.

[63] Yu Chen, Lingfei Wu, and Mohammed Zaki. Iterative deep graph learning for graph neural networks: Better and robust node embeddings. In *NeurIPS*, 2020.

[64] Chuizheng Meng, Sam Griesemer, Defu Cao, Sungyong Seo, and Yan Liu. When physics meets machine learning: A survey of physics-informed machine learning. *Machine Learning for Computational Science and Engineering*, 1 (1):1–23, 2025.

[65] Yassine Zniyed, Thanh Phuong Nguyen, et al. Enhanced network compression through tensor decompositions and pruning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3):4358–4370, 2024.

[66] Denis Lukovnikov, Jens Lehmann, and Asja Fischer. Improving the long-range performance of gated graph neural networks. *arXiv preprint arXiv:2007.09668*, 2020.

[67] Ruoyu Li, Sheng Wang, Feiyun Zhu, and Junzhou Huang. Adaptive graph convolutional neural networks. In *Proc. of AAAI*, pages 3546–3553, 2018.

[68] Hoang Nguyen and Takanori Maehara. Graph homomorphism convolution. In *ICLR*, pages 7306–7316, 2020.

[69] L. Hermansson, F. D. Johansson, and O. Watanabe. Generalized shortest path kernel on graphs. In *International Conference on Discovery Science*, pages 78–85, 2015.

[70] U Kang, Hanghang Tong, and Jimeng Sun. Fast random walk graph kernel. In *Proceedings of the Twelfth SIAM International Conference on Data Mining, Anaheim, California, USA, April 26-28, 2012*, pages 828–838, 2012.

[71] S. Kiefer and P. Schweitzer. Upper bounds on the quantifier depth for graph differentiation in first-order logic. In *Symposium on Logic in Computer Science*, pages 287–296, 2016.

[72] Zhen Zhang, Mianzhi Wang, Yijian Xiang, Yan Huang, and Arye Nehorai. RetGK: Graph kernels based on return probabilities of random walks. In *NeurIPS*, pages 3968–3978, 2018.

[73] Jiaxuan You, Rex Ying, and Jure Leskovec. Position-aware graph neural networks. In *ICLR*, pages 7134–7143, 2019.

[74] Liangyue Li, Hanghang Tong, Yanghua Xiao, and Wei Fan. *Cheetah*: Fast graph kernel tracking on dynamic graphs. In *Proceedings of the 2015 SIAM International Conference on Data Mining, Vancouver, BC, Canada, April 30 - May 2, 2015*, pages 280–288, 2015.

[75] I. Guyon. Design of experiments for the NIPS 2003 variable selection benchmark, 2003.

[76] Yiru Pan, Xingyu Ji, Jiaqi You, Lu Li, Zhenping Liu, Xianlong Zhang, Zeyu Zhang, and Maojun Wang. Csgdn: Contrastive signed graph diffusion network for predicting crop gene-phenotype associations. *arXiv preprint arXiv:2410.07511*, 2024.

[77] Sandra Kiefer. The weisfeiler-leman algorithm: an exploration of its power. *ACM SIGLOG News*, (3):5–27, 2020.

[78] Martin Grohe. The descriptive complexity of graph neural networks. 2023.

[79] Federico Monti, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M Bronstein. Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673*, 2019.

[80] Fredrik D. Johansson and Devdatt P. Dubhashi. Learning with similarity functions on graphs using matchings of geometric embeddings. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery*

and Data Mining, Sydney, NSW, Australia, August 10-13, 2015, pages 467–476, 2015.

[81] Simon S. Du and Jason D. Lee. On the power of over-parametrization in neural networks with quadratic activation. In *ICLR*, 2018.

[82] Xiner Li, Shurui Gui, Youzhi Luo, and Shuiwang Ji. Graph structure extrapolation for out-of-distribution generalization. In *ICML*, 2024.

[83] Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, et al. Exploring the potential of large language models (llms) in learning on graphs. *ACM SIGKDD Explorations Newsletter*, 25(2):42–61, 2024.

[84] Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature. *arXiv preprint*, 2021.

[85] Ilan Newman and Christian Sohler. Every property of hyperfinite graphs is testable. In *Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011*, pages 675–684, 2011.

[86] Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, William L. Hamilton, David Duvenaud, Raquel Urtasun, and Richard S. Zemel. Efficient graph generation with graph recurrent attention networks. In *NeurIPS*, 2019.

[87] Tian Xie, Arthur France-Lanord, Yanming Wang, Jeffrey Lopez, Michael A Stolberg, Megan Hill, Graham Michael Leverick, Rafael Gomez-Bombarelli, Jeremiah A Johnson, Yang Shao-Horn, et al. Accelerating amorphous polymer electrolyte screening by learning to reduce errors in molecular dynamics simulated properties. *Nature communications*, 13(1):3415, 2022.

[88] Maxime Gasse, Didier Chételat, Nicola Ferroni, Laurent Charlin, and Andrea Lodi. Exact combinatorial optimization with graph convolutional neural networks. In *NeurIPS*, 2019.

[89] Nadav Dym, Haggai Maron, and Yaron Lipman. Ds++: A flexible, scalable and provably tight relaxation for matching problems. In *ACM SIGGRAPH ASIA*, 2017.