Position: Graph Foundation Models are Already Here

Haitao Mao [*1] **Zhikai Chen** [*1] **Wenzhuo Tang** [1] **Jianan Zhao** [2 3] **Yao Ma** [4] **Tong Zhao** [5] **Neil Shah** [5]
Mikhail Galkin [6] **Jiliang Tang** [1]

Abstract

Graph Foundation Models (GFMs) are emerging as a significant research topic in the graph domain, aiming to develop graph models trained on extensive and diverse data to enhance their applicability across various tasks and domains. Developing GFMs presents unique challenges over traditional Graph Neural Networks (GNNs), which are typically trained from scratch for specific tasks on particular datasets The primary challenge in constructing GFMs lies in effectively leveraging vast and diverse graph data to achieve positive transfer. Drawing inspiration from existing foundation models in the CV and NLP domains, we propose a novel perspective for the GFM development by advocating for a "graph vocabulary", in which the basic transferable units underlying graphs encode the invariance on graphs. We ground the graph vocabulary construction from essential aspects including network analysis, expressiveness, and stability. Such a vocabulary perspective can potentially advance the future GFM design in line with the neural scaling laws. All relevant resources for GFMs design can be found at here.

1. Introduction

Foundation models (Bommasani et al., 2021), which are pre-trained on massive data and can be adapted to tackle a wide range of downstream tasks, have achieved inimitable success in various domains, e.g., computer vision (CV) (Radford et al., 2021) and natural language processing (NLP) (Bubeck et al., 2023; Touvron et al., 2023). Typically, foundation models can effectively utilize both the prior knowledge obtained from the pre-training stage and the data from downstream tasks to achieve better performance (Han et al., 2021) and even deliver promising efficacy with few-shot task demonstrations (Dong et al., 2022; Mao et al., 2024).

Meanwhile, graphs are vital and distinctive data structures that encapsulate non-Euclidean and intricate object relationships. Since various graphs embody unique relations, most graph learning approaches are tailored to train from scratch for a single task on a particular graph. This approach necessitates separate data collection and deployment for each individual graph and task. Consequently, an intriguing question emerges: *Is it possible to devise a Graph Foundation Model (GFM) that benefiting from large-scale training with better generalization across different domains and tasks?*

Despite advanced foundation models in other domains, the development of GFMs remains in the infant stage. Recent research has demonstrated initial successes of GFMs in specialized areas, such as knowledge graphs (Galkin et al., 2023; 2024) and molecules (Beaini et al., 2023). Notably, most of these models are built on principles specific to their domains. For instance, ULTRA for knowledge graph completion (Galkin et al., 2023) draws inspiration from double equivariance for inductive link prediction (Gao et al., 2023a). However, there is still a lack of general guidance on how to build GFMs that can effectively cater to a broad spectrum of graph-based applications.

The key difficulty in designing GFMs lies in finding the invariance across diverse graph data, ranging from social networks to molecular graphs with countless structural patterns, into the same representation space to achieve positive transfer. The answer from the CV and NLP domains is a shared vocabulary. In the NLP foundation models, the text is first broken down into smaller units based on the vocabulary, which can be words, phrases, or symbols. In the CV foundation models, the image is mapped to a series of discrete image tokens (Yu et al., 2023; Bai et al., 2023) based on the vision token vocabulary. The vocabulary defines the basic units in the particular domain, transferable across different tasks and datasets. Therefore, the key challenges in achieving the GFM narrow down to how we can find the **graph vocabulary**, the basic transferable units underlying graphs to encode the invariance on graphs.

[*]Equal contribution [1]Michigan State University, East Lansing, US [2]Université de Montréal [3]Mila - Québec AI Institute [4]Rensselaer Polytechnic Institute, Albany, US [5]Snap Inc., USA. [6]Intel Labs. Correspondence to: Haitao Mao <haitaoma@msu.edu>, Zhikai Chen <chenzh85@msu.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

However, finding a suitable graph vocabulary that works across diverse graphs is challenging, which is the primary focus of this paper.

Our contributions: In this paper, we present a vocabulary perspective to clearly state the position of the GFM. In particular, we attribute the existing success of primitive GFMs to the suitable vocabulary construction guided by the particular transferability principle on graphs in Section 2. A comprehensive review of the graph transferability principles and corresponding actionable steps is illustrated in Section 3, serving us the principle for future vocabulary construction and the GFM design. In Section 4, we discuss the potential for building the GFM following neural scaling laws from several perspectives (1) building and training vocabulary from scratch, and (2) leveraging existing LLM. Finally, we introduce more insights and open questions to inspire constructive discussions on the GFM in Section 5.

2. Existing GFMs and Key Designs

Existing GFMs (Galkin et al., 2023; Zheng et al., 2023a) have achieved initial success, including promising zero-shot generalization to unseen graphs. Based on model transferability, current GFMs can be categorized into task-specific, domain-specific, and primitive GFMs. Definitions for all categories can be found in Section 2.1. The key to a successful GFM design is further discussed in Section 2.2. Notably, none of the current GFM have the capability to transfer across all graph tasks and datasets from all domains, despite such expectations being achieved in the NLP domain (Bubeck et al., 2023; Touvron et al., 2023) with long-term effort. GFMs remain in a nascent stage with limited development. Despite the gap compared to the success in the NLP domain, GFMs have already achieved significant improvement over existing GNNs with end-to-end training on a single dataset. However, the feasibility of general GFM remains unclear with unique graph challenges. Graphs are abstract data structures which are more diverse than natural language text and images grounded in the physical world

2.1. Existing GFM Categories

Based on the model transferability across domains and tasks, we can roughly distinguish the existing primitive GFMs into three categories: task-specific, domain-specific, and primitive GFMs. We provide definitions and examples for each category, with a more comprehensive illustration in Appendix B.

A *task-specific/domain-specific GFM* should be transferable across the specific task/domain and thus adapt to diverse downstream datasets and domain-specific tasks. A notable example of a task-specific GFM is ULTRA (Galkin et al., 2023), achieving superior zero-shot knowledge graph completion performance across datasets from various domains. A task-specific GFM shows great practical benefits, as it can be trained on data-rich domains, e.g., Wikipedia knowledge graphs, and subsequently improve effectiveness in resource-limited domains, e.g., geography knowledge graph. A domain-specific GFM instance, DiG (Zheng et al., 2023a), learns universal representations across various chemical tasks by leveraging domain-specific knowledge. The domain-specific GFM is highly efficient, as one model can serve all tasks while also delivering improved effectiveness compared to single-task models.

A *primitive GFM* exhibits the capability to generalize towards a limited number of datasets and tasks. A notable example is OFA (Liu et al., 2023b), which is co-trained on data ranging from citation networks and molecule graphs to knowledge graphs via a unified task formation on node, link, and graph level tasks. The OFA model can achieve comparable or even better performance over the vanilla GNNs on each task. Nonetheless, OFA requires transforming all node features into text for co-training, which may not be convenient for all types of data. This co-training paradigm may also limit its generalization to unseen tasks and domains.

2.2. The Key to A Successful GFM Design.

Despite the empirical success achieved by existing GFMs, most of them are inspired by domain/task-specific principles. In this section, we aim to illustrate the common design approach using ULTRA (Galkin et al., 2023) as a showcase.

ULTRA (Galkin et al., 2023) is a task-specific GFM focusing on the knowledge graph completion (KGC) task. The KGC task aims to infer the missing triplet (edge), denoted as (h, r, t), where r is a query relationship, h and t are the head and tail entities, respectively. The KGC model aims to answer the query $(h, r, ?)$ by predicting the tail entity t.

The first reason for its success is to utilize the NBFNet (Zhu et al., 2021b) backbone model which enables the inductive generalization to new graphs with an expressive relational vocabulary. The NBFNet proposes a conditional message passing that can learn the pairwise-node representation conditioned on a head entity node and a query relation.

Huang et al. (2023c) demonstrates that this conditional message passing, grounded in the relational Weisfeiler-Leman algorithm, theoretically offers greater expressiveness in KGC compared to standard, unconditional GNNs (Li et al., 2022). Such expressiveness helps to distinguish the difference between knowledge graphs with different structural features, leading to a suitable relational vocabulary. In contrast, Barcelo et al. (2022) indicates that those unconditional GNNs, e.g., R-GCN (Schlichtkrull et al., 2018) and CompGCN (Vashishth et al., 2019), map non-isomorphic node pairs into the same representation, leading to a con-

tracted relational vocabulary. Such contracted vocabulary may lead to negative transfer with inappropriately generalizing knowledge across non-isomorphic node pairs with inherent differences.

However, such expressive relational vocabulary only considers the pre-defined relation types which cannot generalize to the scenario with new relation types during inference. To extend the existing relational vocabulary including new relationship type, Galkin et al. (2023) constructs a graph of relations that captures fundamental interactions independent from any graph-specific relation types, serving as the second reason for its success. The graph of relations is theoretically grounded (Gao et al., 2023b) which aims to learn the double permutation-equivariant representations. Such representation is equivariant to permutations of both node entities and edge relation types. Such equivariance can be an analogy to a shared relational vocabulary. It connects the new unseen relationship types to the existing ones and maps the equivariant node pairs into the same representation despite different relation types, leading to the positive transfer.

In summary, we can conclude the key for ULTRA to achieve good transferability is finding a suitable vocabulary for KGC satisfying two principles: (1) The vocabulary should not be compacted, which causes distinct node pairs to share representations, leading to potential negative transfer. (2) The vocabulary should be sufficiently inclusive to map new, unknown relationships onto the existing vocabulary, potentially enabling positive transfer. Notably, the vocabulary design in GFMs does not necessarily correspond to a tokenizer or an embedding layer as in the NLP domain. Instead, it can involve a model that maps graphs from different domains into the same representation space, enabling positive transfer and serving as a prerequisite for data-scaling.

The effectiveness of finding a suitable vocabulary for building the GFM can also be found in other existing primitive GFMs with the following evidence. GraphGPT (Zhao et al., 2023b) constructs a dataset-specific vocabulary where each node corresponds to a unique node ID. Notably, GraphGPT requires specific pre-training and fine-tuning on each dataset. MoleBERT (Xia et al., 2023), the foundation model for molecule graphs, manually designs a vocabulary that transforms atom attributes into chemically meaningful codes.

3. Graph Transferability Principles with Actionable Steps

In the last section, we investigate the key to building an effective GFM, which lies in constructing a suitable graph vocabulary to keep the essential invariance across datasets and tasks. Despite existing successes, more graph transferability principles, identifying different invariances, can serve as guidance for constructing new suitable graph vocabulary for future GFMs. We present a few actionable next steps inspired by these principles, highlighting their potential benefits.

The following discussions are organized as follows: We first provide a general introduction to the graph transferability principles in Section 3.1. Detailed task-specific principles on node classification, link prediction, and graph classification tasks can be found in Section 3.2, 3.3, and 3.4, respectively. We finally discuss the principles for task transferability in Section 3.5. Notably, the following discussions majorly concentrate on the transferability of the graph structure. The discussion about techniques for aligning the feature space can be found in Appendix C.

3.1. An overview on Graph transferability principles

In this subsection, we introduce principles that enable transferability on graphs, focusing on three key aspects: network analysis, expressiveness, and stability. More discussion on other principles revolving on deeper GNNs can be found in Appendix D.

Network analysis provides a conventional understanding of the network system by identifying fundamental graph patterns, e.g., network motif (Menczer et al., 2020) and establishing the key principles, e.g., triadic closure principle (Huang et al., 2015) and homophily principle, which are generally valid across different domains. Those principles have been generally utilized to guide the design of advanced GNNs. For example, the state-of-the-art GNN for link prediction (Wang et al., 2023b) is a Neural Common Neighbor, inspired by the triadic closure principle. Despite its effectiveness, network analysis heavily relies on expert knowledge without a provable guarantee.

Expressiveness provides a theoretical background as to which functions graph neural architectures can model in general, e.g., a well-known connection that graph-level performance of GNNs is bounded by Weisfeiler-Leman tests (Xu et al., 2019; Morris et al., 2019; 2023). The most-expressive structural representation (Srinivasan & Ribeiro, 2019) is the key concept describing that the representation of two node sets should be invariant if and only if the node sets are symmetric with a permutation equivalence. Such most-expressive structural representation serves as an important principle to design a suitable graph vocabulary that perfectly distinguishes all non-isomorphic structural patterns in multi-ary prediction tasks.

Stability (Ruiz et al., 2023) assesses the representation sensitivity to graph perturbations. It aims to maintain a bounded gap in predictions for pairs under minor perturbations, rather than the expressiveness only distinguishing between isomorphic and non-isomorphic cases. The stability imposes a stricter constraint leading to better generalization. It can be

an analogy to the constraint on the graph vocabulary where similar structure patterns should have similar representation.

3.2. Transferability Principles in Node Classification

Network analysis. *Homophily* (Khanam et al., 2020), which describes the phenomenon of linked nodes often sharing similar features ("birds of a feather flock together"), is a longstanding principle in social science. It serves as the principle guidance for methods ranging from conventional pagerank (Chien et al., 2021) and label propagation (Chawla & Karakoulas, 2005) to the recent advanced GNNs. Existing GNN architectures, often crafted based on the homophily principle, demonstrate strong performance on diverse homophilous graphs across various domains. This adherence to homophily not only enhances model effectiveness but also facilitates model transferability among homophilous graph datasets. Notably, successful transfers among such graphs are evidenced in Ying et al. (2018).

While homophily predominates in network analysis, it is not a universal rule. In many real-world scenarios, "opposites attract", resulting in networks characterized by heterophily—where nodes are more likely to link with dissimilar nodes. GNNs built with the homophily principle often struggle with heterophilious networks, except in cases of "good heterophily" (Ma et al., 2021; Luan et al., 2021), where GNNs can identify and leverage consistent patterns in connections between dissimilar nodes. However, most heterophilious networks are complex and varied, posing challenges for GNNs due to their irregular and intricate interaction patterns (Luan et al., 2023; Wang et al., 2024a; Mao et al., 2023a). Consequently, GNNs' transferability, more assured in homophilous graphs, is facing significant challenges in heterophilous ones.

Stability. You et al. (2023) theoretically establishes the relationship between transferability and network stability, demonstrating that graph filters with enhanced spectral smoothness and a smaller maximum frequency response exhibit improved transferability in terms of node features and structure, respectively. In particular, spectral smoothness, characterized by the Lipschitz constant of the graph filter function of the corresponding GNN, indicates stability against edge perturbations. The maximum frequency response, reflecting the highest spectral frequency after applying a graph filter (essentially the largest eigenvalue of the Laplacian matrix), describes stability against feature perturbations.

Actionable steps inspired by principles. (Mao et al., 2023a) illustrates the network analysis principle that a single GNN can perform well on either homophily patterns or heterophily patterns, but not both. This principle provides the actionable insight for GFM design, suggesting that the graph vocabulary for homophily patterns and heterophily patterns should be modeled separately. Consequently, the model backbone for GFMs in node classification should not rely on a single GNN, which only excels on either homophilic graphs or heterophilic graphs. A better architecture design choice could be (1) an adaptive GNN with different aggregation filters for homophilic and heterophilic graphs, or (2) a graph transformer without a fixed aggregation process.

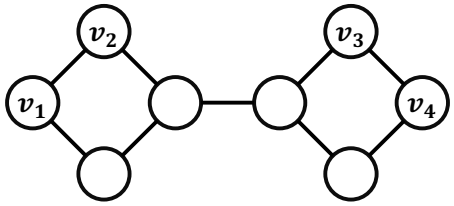

Figure 1. In this graph, nodes v_1 and v_4 are isomorphic; links (v_1, v_2) and (v_2, v_4) are not isomorphic. However, vanilla GNN with the same node representations v_1 and v_4 gives the same prediction to links (v_1, v_2) and (v_2, v_4).

You et al. (2023) designs a spectral regularization term inspired by the network stability to address the out-of-distribution problem. Adapting spectral regularization for GFMs could be a potential next step.

3.3. Transferability Principles in Link Prediction

Network Analysis. Important network analysis principles (Mao et al., 2023b) fall into three primary concepts including: (1) local structural proximity corresponding to the triadic closure principle (Huang et al., 2015), where friends of friends become friends themselves. It inspires well-known conventional methods including CN, RA, AA (Adamic & Adar, 2003). (2) global structural proximity corresponding to the decay factor principle, where two nodes with more short paths between them have a higher probability of being connected. It inspires well-known conventional methods e.g., Simrank and Katz (Katz, 1953; Jeh & Widom, 2002). (3) feature proximity corresponding to the homophily principle (Murase et al., 2019) where shared beliefs and thoughts can be found in connected individuals.

These principles guide the evolution of link prediction algorithms, from basic heuristics to sophisticated GNNs (Chamberlain et al., 2022; Li et al., 2023a). GNNs, inspired by these principles, perform well across diverse graphs in multiple domains. Moreover, Zheng et al. (2023b) provides empirical evidence supporting the beneficial transferability of these guiding principles.

Expressiveness. A vanilla GNN, equipped with only single-node permutation equivalence, cannot achieve transferability for the link prediction task due to its lack of expressiveness. An example to showcase such failure is shown in Figure 1 with a featureless graph. v_1 and v_4 are represented identically by the vanilla GNN, as they possess identical

neighborhood structures.

Therefore, the similarity between v_1 and v_2 will be the same as the one between v_4 and v_2, leading to identical representations and predictions for both links (v_1, v_2) and (v_2, v_4) However, according to the global structural proximity, (v_1, v_2), with a shorter distance of 1, should be more likely to be connected. The vanilla GNN, computing v_1's representation solely from its neighborhood, overlooks the structural dependence with v_2. As a result, this potentially leads to negative transfer, where the GNN might erroneously predict both or neither link to exist, whereas it's more likely that only (v_1, v_2) has a link.

To consider all the possible dependencies between node pairs, we aim for the most expressive structural representation for the link prediction. This representation should be invariant if and only if links are symmetric. Zhang & Chen (2018) achieves such structural representation by incorporating node labeling features that depend on both the source and target nodes in a link. Zhang et al. (2021) further highlights the key aspects of node labeling design, including: (1) target-nodes-distinguishing, where the source and target nodes have distinct labels compared to other nodes; and (2) permutation equivariance. Node labeling methods that fulfill these criteria, such as double radius node labeling (DRNL) and zero-one (ZO) labeling, can produce the most expressive structural representations. Many other GNNs (You et al., 2021; 2019; Wang et al., 2021) can achieve similar expressiveness, serving as the potential backbone for GFM on the link prediction task. The expressiveness representation can find the complete set of distinct relations to differentiate all non-isomorphic node pairs, thereby mitigating the risk of negative transfer in standard GNNs. Huang et al. (2023c) extends the relational Weisfeiler-Leman framework (Barcelo et al., 2022) to link prediction and incorporate the concept of labeling tricks to multi-relational graphs.

Stability. For those equally expressive structural representations, there may still be a gap in terms of their stability. For example, empirical evidence (Zhang et al., 2021) shows that GNNs with DRNL labeling outperform those with ZO labeling. From the perspective of stability, it is crucial to maintain a bounded gap in predictions for pairs under minor perturbations. Wang et al. (2021) provides a theoretical analysis identifying key properties of stable positional encoding (GNNs should be rotation and permutation equivariant to positional encodings) that enhance generalization. The stable positional encoding may be directly applied towards better GFMs.

Actionable step inspired by principles. (Mao et al., 2023b) illustrates the network analysis principle concerning the incompatibility between structural proximities and feature proximity. Node pairs with high feature proximity are likely to be with low local structural proximity and vice versa. This incompatibility leads to over-emphasis on node pairs with high structural proximity while neglecting those with high feature proximity. This principle provides actionable insight for GFM design, suggesting that the graph vocabulary for feature proximity patterns and structural proximity patterns should be modeled separately. Consequently, the model backbone for GFMs in link prediction should separately encode the pairwise structural proximity and the feature proximity.

A GNN following the expressiveness principles could include all the important structural information relevant to the link prediction (Zhang et al., 2021). Dong et al. (2024) utilizes in-context learning to effective transfer expressive GNN representations to new, unseen graphs. Satisfying performance can be found across graphs from biology, transport, web, and social domains. An actionable next step could be to better utilize expressive representations for downstream graphs from specific domains.

3.4. Transferability Principles in Graph Classification

Network Analysis. Network motifs, typically composed of small and recurrent subgraphs, are often considered the building blocks of a graph (Milo et al., 2002; Benson et al., 2016). A proper selection of the motif set can cover most essential knowledge on the specific datasets. Graph kernels (Vishwanathan et al., 2010) are proposed to quantify motif counts or other pre-defined graph structural features and then utilize the extracted features to build a classifier such as SVM. Despite the essential motif sets from different domains being generally different, there could exist a uniform set of motifs shared across different domains. In such cases, the positive transfer can be found on the uniform sets, where Battiston et al. (2020) shows the positive transfer across neuronal connectivity networks, food webs, and electronic circuits. Therefore, we conjecture that the network motif could be the base unit for the vocabulary (a set of invariant elements) for the graph classification as it is both explainable and potentially shared across graphs.

Expressiveness. Zhang et al. (2024) proposes a unified framework to understand the ability of different GNNs to detect and count graph substructures (motif). More expressive GNN which could detect more diverse motifs and construct a richer graph vocabulary. In analogy with the uniform motif sets, we conjecture that it is more possible for the expressive GNN to find the uniform motif sets and achieve better transferability.

Stability. Huang et al. (2023d) proposes a provably stable position encoding that surpasses the expressive sign and invariant encoding (Kreuzer et al., 2021) and modeling (Lim et al., 2022), enabling minimal changes to positional encodings on the minor modifications to the Laplacian. The key innovation is to apply a weighted sum of eigenvectors in-

stead of treating each eigensubspace independently. Satisfying performance can be observed on the out-of-distribution molecular graph prediction. Such stable positional encoding may be directly applied towards better GFMs.

Actionable step inspired by principles. Inspired by the network analysis with graph kernels, one concrete next step towards GFMs could be revolving on how to identify frequent network motif (Hočevar & Demšar, 2014; Ribeiro et al., 2021) which should be transferable across all graphs. Expressive GNNs with better network motif model capability could be a suitable architecture towards GFMs.

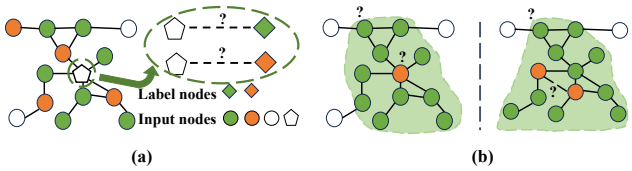

Figure 2. Unifying different task formulations: (a) Link view: Given the target node, node classification is converted to the link prediction between the target node and corresponding label nodes. (b) Subgraph view: Node classification (orange node) is converted to the (green) ego-graph classification. Link prediction (orange nodes) is converted to the (green) induced-subgraph classification.

3.5. Transferability Principles across Tasks

A unified task formulation is generally employed to facilitate transferability across various tasks. The unified task formulation enables (1) enlarging the dataset size via converting datasets for different downstream tasks as one and (2) utilizing one pre-training model to serve different tasks. The significance of aligning task formulations is evident in the following example: Jin et al. (2020b) shows that using link prediction directly as a pretext task leads to negative transfer for node classification. However, by reformulating node classification into a link prediction problem (Sun et al., 2022; Huang et al., 2023a), where a node's class membership is treated as the link likelihood between the node and label nodes, positive transfer is achieved. Liu et al. (2023f); Sun et al. (2023) further propose a sub-graph view to adapt the node classification as an ego graph classification, and link prediction as a binary classification on the induced subgraph of the target node pair. Figure 2 provides illustrative examples for these two unified views. More recently, Liu et al. (2023b) unifies node-level, link-level, and graph-level tasks via (1) adding a virtual prompt node and (2) connecting the virtual nodes to nodes of interests, i.e., the center node for node classification, source and target nodes for link prediction, and all nodes for graph classification.

A unified formulation provides the possibility for co-training all tasks together while it remains unknown whether this can be done without negative transfer. Moreover, the unified task formulation may not be necessary to achieve transfer across tasks. It is generally utilized for supervised co-training and prompt-based prediction as discussed above. A GFM can be (1) pre-trained with self-supervised tasks and (2) adapted to downstream tasks via fine-tuning without requiring specific task formulations. The success is due to the transferability principles across different tasks. However, there remains limited study in this direction. We list a few existing principles as follows. (1) Node classification and link prediction tasks share the feature homophily as an important principle. (2) Liu et al. (2023c) indicates that the global structural proximity principle on the link prediction can improve the node classification performance on the non-homophilous graph. (3) The triadic closure in the link prediction is a particular network motif utilized in the graph classification. There are more shared motifs (Hibshman et al., 2021; Dong et al., 2017; AbuOda et al., 2020; Kriege et al., 2020) on both graph classification and link prediction tasks. We emphasize the importance of cross-task transferability principles as an important future direction.

4. Neural Scaling Law on GFMs

The success of the foundation model can be attributed to the validity of the neural scaling law (Kaplan et al., 2020) which shows performance enhancement with increasing model scale and data scale. In this section, we first discuss when the neural scaling law happens in Section 4.1 from a graph vocabulary perspective. We then discuss techniques towards successful data scaling and model scaling in Section 4.2 and 4.3, respectively. We finally discuss the potential on leveraging large-scale LM on the graph domain in Section 4.4. More discussions on technical details can be found in Appendix C.

4.1. When Neural Scaling Law Happens

In section 3, we discuss the underlying transferable principles guiding future vocabulary construction. Such principle guidance has led to the successful scaling behavior in the material science domain (Shoghi et al., 2023; Zhang et al., 2023a; Batatia et al., 2023) with the help of the geometric prior. Nonetheless, we are still cautious about whether the existing success can be extended to the graph domain. The key concern is whether graphs can strictly follow those principles. Uncertainty can be found on the human-defined graph construction criteria (Brugere et al., 2018). For instance, the construction knowledge relying on expert knowledge may lead to uncertainty in edges (Ye et al., 2022). Chen et al. (2023b); Li et al. (2023c) observe that the mislabeled samples widely exist across datasets, where the popular CITESEER dataset has more than 15% wrongly labeled data. Despite the above uncertainty, different graph constructions with manual design can follow opposite principles. For instance, OGBN-ARXIV (Hu et al.,

2020) and ARXIV-YEAR (Lim et al., 2021) are two node classification datasets with identical graph information. The only difference lies in the label where OGBN-ARXIV employs paper categories, and ARXIV-YEAR uses publication years as labels, resulting in conflicting homophily and heterophily properties (Mao et al., 2023a). Therefore, when uncertainties and opposite graph constructions exist, the scaling behavior may not happen as the data does not obey the graph transferable principles.

4.2. Data Scaling

Data scaling refers to the phenononmon that the performance consistently improves with the increasing data scale. Chen et al. (2023a); Huang et al. (2023a) initially validate that GNNs trained in both supervised and self-supervised manners follow data scaling law on molecular property predictions, and node classification on text-attributed graphs. Cao et al. (2023) further exhibits that the similarity between pre-training data and downstream task data serves as a prerequisite for the data scaling on graphs. Specifically, Cao et al. (2023) provides concrete guidance on how to select the pre-training data via the graphon signal analysis and the essential network property, i.e. network entropy, respectively. Notably, all principles mentioned in Section 3 can be applied to facilitate positive transfer with data scaling phenomena.

A limitation in current research on data scaling is graph data insufficiency, in contrast to the readily available trillion-level real-world data in CV and NLP domains. The key reasons are two-fold: (1) constructing graphs requires expert intervention e.g., defining relationships (2) intellectual property issues. We endeavour to collect all the open-source graph datasets with details in Appendix A.

Synthetic graph generation can be utilized to alleviate the data insufficiency issue, enableing more comprehensive training. Traditional graph generative models (Albert & Barabási, 2002; Robins et al., 2007; Airoldi et al., 2008; Leskovec et al., 2010) are capable of generating graphs satisfying some certain statistical properties, which still plays an important role on node-level and link-level tasks. Deep generative models on graph (Jin et al., 2020a; Luo et al., 2021; Jo et al., 2022; Vignac et al., 2023; Liu et al., 2023a) have shown great success in generating high-quality synthetic graphs which helps graph-level tasks by providing a more comprehensive description of the graph distributions space. With successful evidence of pre-training on synthetic data from other domains (Mishra et al., 2022; Trinh et al., 2024), we anticipate the potential on benefits from high-quality synthetic graphs.

4.3. Model Scaling

Model scaling refers to the phenomenon that the performance consistently improves with the increasing model scale. Previous research in NLP indicates that apart from data, the backbone model constitutes a fundamental for scaling (Kaplan et al., 2020). Liu et al. (2024a) primarily validates the neural scaling law on various graph tasks and model architectures under the supervised setting.

However, Kim et al. (2022) demonstrates that the GAT (Veličković et al., 2017) with a larger number of parameters underperforms on the graph regression tasks compared to the smaller-sized counterparts. As a comparison, geometric GNNs scale well to predict atomic potentials in material science (Shoghi et al., 2023; Zhang et al., 2023a; Batatia et al., 2023). Observations indicate that geometric GNNs with a good geometric-prior vocabulary design can help achieve model scaling over the vanilla GNN.

Graph transformer is another popular choice for the model architecture, where geometric-prior graph vocabulary design is explicitly modeled through either a GNN encoder or positional encoding (Müller et al., 2023). Masters et al. (2022); Lu et al. (2023) show that graph transformers show positive scaling capabilities for molecular data under a supervised setting. More recently, Zhao et al. (2023b) demonstrates vanilla transformer's effectiveness in protein and molecular property prediction. Particularly, it views the graph as a sequence of tokens forming an Eulerian path (Edmonds & Johnson, 1973), which ensures the lossless serialization, and then adopts next-token prediction to pre-train transformers. After fine-tuning, it achieves promising results on the protein association prediction and molecular property prediction and shows that vanilla transformers also follow the model scaling law (Kaplan et al., 2020). Nonetheless, the effectiveness of transformers on other tasks remains unclear.

4.4. Leveraging Large-scale LMs for Graphs

LLMs with successful scaling behavior have achieved tremendous success in the NLP domain. Surprisingly, well-trained LLMs can be applied to other domains with satisfying performance such as time series forecast (Gruver et al., 2024a) and material science (Gruver et al., 2024b). Larger-scale LLMs can even capture key symmetries of crystal structures, suggesting that LLMs may posses a strong simplicity bias (Panwar et al., 2023) across domains by implementing Bayesian model averaging algorithm (Zhang et al., 2023c).

A recent line of research on GFMs focuses on leveraging strong capabilities of LLMs on graph tasks. Our discussions can be roughly categorized on LLM applications (i) conventional graph tasks (such as node, edge, and graph classification), and (ii) language-driven tasks like Graph

Question Answer (GQA).

LLMs on conventional graph tasks. One natural way to utilize LLMs is as textual feature encoders (Chen et al., 2023b). Despite original node features may not be text, Liu et al. (2023b) manually converts them into knowledge-enhanced text descriptions and then encodes features into textual embedding. This LLM embedding approach offers the following benefits. (i) High feature quality helps achieve satisfying performance with vanilla GCN (Chen et al., 2023b). (ii) LLMs encode diverse original features into an aligned feature space, enabling training and inference across graphs from different domains without the feature heterogeneity problem. Notably, when LLMs are utilized as feature encoders for textural understanding, the scaling law does not happen (BehnamGhader et al., 2024), meaning that a larger model does not necessarily lead to better performance.

Another approach is to utilize LLMs as predictors which first fine-tunes LLM and then generate predictions in a natural language form. Chen et al. (2023b); He et al. (2023) treats node classification as text classification on the target node feature, illustrating promising results in the zero-shot setting. However, simply flattening graph structures into prompts does not yield additional improvement, remaining a large performance gap compared to well-trained GNNs (Chen et al., 2023b). To better encode the graph structure knowledge, methods such as GNN (Tang et al., 2023), graph transformer (Chai et al., 2023), and non-parametric aggregation (Chen et al., 2024b) are utilized as structure encoders. The encoded structural embeddings are then linearly mapped into text space as prompt tokens. LLMs generate predictions based on a concatenation of the prompt token and the textual instruction. Instead of additional graph modeling, Zhao et al. (2023a) employs a novel tree-based prompt design that transforms the graph into sequence while retaining important structural semantics. This approach indicates the potential for LLMs to understand particular graph structures. Overall speaking, a proper LLM fine-tuning can achieving satisfying graph performance while the efficiency may be a potential issue.

LLMs on language-driven graph tasks. Instead of adapting LLM for conventional graph tasks, LLMs can also be applied to language-driven tasks they originally skilled in, for example, Graph Question Answer (GQA). Fatemi et al. (2023); Wang et al. (2023a) apply LLMs on various GQA tasks, e.g, cycle check, and maximum flow, by describing graph structure with natural language. More recently, Perozzi et al. (2024) incorperates an external GNN tokenizer to encode graph information, achieving satisfying out-of-domain generalization to unseen graph tasks. Interestingly, Perozzi et al. (2024) illustrates that equivariance is not necessary when equipped with LLMs. (He et al., 2024) proposes new real-world challenging GQA tasks and a corresponding LLM-based conversational framework. This framework integrates GNNs and retrieval-augmented generation (RAG) to improve graph understanding and mitigate issues like hallucination, demonstrating effectiveness across multiple domains. Until now, most GQA challenges have focused on the abstract graphs without concrete descriptions for each node, creating to obstacle to o leveraging the extensive internal knowledge in LLMs. We call for more real-world GQA challengesm enabling better leverage LLM capabilities.

Despite the above successes, it remains concerns on the LLM's capability on understanding the essential graph structures. Saparov & He (2022); Dziri et al. (2023) theoretically observe that the LLM is required to tackle problems sequentially greedily (McCoy et al., 2023), leading to a shortcut solution rather than a formal analysis on the graph structure. A more comprehensive discussion can be found in Appendix E.1.

5. Insights & Open Questions

In this section, we explore key insights gained from recent advancements in GFMs and highlight open questions that remain to be addressed in this evolving field. More comprehensive discussions can be found in Appendix E

5.1. Potential Redundancy on Pretext Task and Architecture Design

There are mainly two approaches to achieving transferability: (1) designing GNNs with specific geometric properties for transfer, e.g., ULTRA (Galkin et al., 2023), and (2) creating pretext tasks to automatically learn these properties. (Jin et al., 2020b) suggests an overlap between these approaches, indicating that pretext tasks targeting local structural information might be unnecessary, given that GNNs often inherently encode this information. Investigating the strengths and limitations of these techniques, along with providing practical guidance for their selection, could be a valuable research direction. A hypothesis might be that model design methods are more suitable for data that strictly adheres to geometric priors, while pretext task designs are more effective in the opposite scenario.

5.2. The Feasibility of GFMs

Graphs can be defined in different ways based on different criteria like similarity or influence between node pairs (Brugere et al., 2018). We can then categorize graphs based on the observability of the criteria. The observable graph is unambiguously known, e.g., whether one paper cites another paper in a citation graph. Text and images can also be viewed as a specific case of observable graphs. In contrast, the unobservable ones are manually

conducted with ambiguous descriptions of the relationship, e.g., whether one gene regulates the expression of another in a gene-regulate graph. These graphs may not naturally exist in the world, leading to uncertainty with a lack of invariant principle. It remains unknown whether GFMs can learn shared knowledge while avoiding manually introduced noisy patterns.

There are concerns about the benefit of training a GFM on graphs that are neither from the same domain nor share the same downstream task. On the one hand, it seems that training on them simultaneously shows no positive transfer benefit while increasing the risk of the negative transfer. On the other hand, there may be potential undiscovered transferable patterns that could lead to success. Therefore, we pose an open question whether there exists a universal structural representation space that can benefit all the graph tasks?

5.3. Broader Usage of GFM

In this paper, we majorly focus on building GFM for conventional graph-focused tasks. Notably, graph formulation provides the universal representation ability, which has a broader usage in other domains, e.g., scene graphs for Computer Vision (CV) (Zhai et al., 2023; Zhong et al., 2021), bipartite graphs for linear programming (Chen et al., 2022), and physical graphs for understanding physical mechanisms (Shi et al., 2022). To emphasize more broader usage of GFMs, we illustrate the potential advantaged usage of GFMs over existing foundation models in reasoning, computer vision, and code intelligence domains domains. Details can be found as follows.

Reasoning. Ibarz et al. (2022) proposes a task-specific GFM, focusing on neural algorithmic reasoning tasks. A strong reasoning capability can be found with effectiveness across sorting, searching, and dynamic programming tasks. We argue that this GFM following the theory of algorithmic alignment (Xu et al., 2020) may achieve better reasoning capability than the LLM merely relying on the textual inputs via retrieving concepts co-occur frequently in training data (Prystawski & Goodman, 2023).

Computer Vision. Scene graph is a data structure representing objects, their attributes, and the relationships between them within an image, facilitating CV tasks such as image understanding and visual reasoning. However, current research remains a naive scene graph modeling with vanilla GNNs with more emphasis on image modeling. We argue that the GFM on the scene graph may help to preserve global and local scene-object relationships (Zhai et al., 2024), avoiding the potential conflict or redundancy between multiple objectives which frequently appears on the recent popular Sora model (Brooks et al., 2024).

Code Intelligence. Graphs, e.g., code property graph (Liu et al., 2024b), control flow graph, and program dependency graph, play an important role in code-relevant tasks, e.g., vulnerability detection (Liu et al., 2024b), fault localization (Rafi et al., 2024), and code search (Ling et al., 2021). Compared to sequence-based modeling with LLMs, graphs can provide a complementary perspective on the overlooked essential code attributes such as syntax, control flow, and data dependencies. However, the graph modeling remains naive with unknown transferability across different program languages.

Overall speaking, GFMs demonstrate unique value compared to foundation models in other domains. However, they are limited to applications involving graph structure data. An exciting future topic is how to adaptively combine GFM with other foundation models across different modality towards a powerful Artificial General Intelligence (AGI).

6. Conclusion

From the transferability principles of graphs, we review existing GFMs and ground their effectiveness from a vocabulary view to find a set of basic transferrable units across graphs and tasks. Our key perspectives can be summarized as follows: (1) Constructing a universal GFM is challenging, but domain/task-specific GFMs are approachable with the usual availability of a specific vocabulary. (2) One challenge is developing GFMs following the neural scaling law, which requires more data collection, suitable architecture design, and properly leveraging LLMs. This paper summarizes the current position of GFMs and challenges toward the next step, which may be a blueprint for GFMs to inspire relevant research.

Acknowledgement

We want to thank Yanqiao Zhu at the University of California, Los Angeles, and Yuanqi Du at Cornell University for their constructive comments on this paper.

Haitao Mao, Zhikai Chen, Wenzhuo Tang, and Jiliang Tang are supported by the National Science Foundation (NSF) under grant numbers CNS 2246050, IIS1845081, IIS2212032, IIS2212144, IOS2107215, DUE 2234015, DRL2025244 and IOS2035472, the Army Research Office (ARO) under grant number W911NF-21-1-0198, the National Telecommunications and Information Administration (NTIA), the Home Depot, Amazon Faculty Award, JP Morgan Faculty Award, Microsoft Research, Meta, and SNAP. Yao Ma is supported by the National Science Foundation (NSF) under grant numbers NSF-2406648 and NSF-2406647.

Impact Statements

In this paper, we provide principle guidance for the development of graph foundation models, which can be a pivotal infrastructure empowering diverse applications like nature science and E-commerce. The graph foundation model may reduce the resource consumption associated with training numerous task-specific models. Moreover, it may substantially curtail the requirement for manual annotation, particularly in domains such as molecular property prediction. We anticipate that our contributions will advance the ongoing efforts aimed at developing next-generation graph foundation models with better versatility and fairness.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, pp. 1–3, 2024.

AbuOda, G., De Francisci Morales, G., and Aboulnaga, A. Link prediction via higher-order motif features. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I*, pp. 412–429. Springer, 2020.

Adamic, L. A. and Adar, E. Friends and neighbors on the web. *Social networks*, 25(3):211–230, 2003.

Airoldi, E. M., Blei, D., Fienberg, S., and Xing, E. Mixed membership stochastic blockmodels. *Advances in neural information processing systems*, 21, 2008.

Albert, R. and Barabási, A.-L. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

Bai, Y., Geng, X., Mangalam, K., Bar, A., Yuille, A., Darrell, T., Malik, J., and Efros, A. A. Sequential modeling enables scalable learning for large vision models. *arXiv preprint arXiv:2312.00785*, 2023.

Barcelo, P., Galkin, M., Morris, C., and Orth, M. R. Weisfeiler and leman go relational. In *Learning on Graphs Conference*, pp. 46–1. PMLR, 2022.

Barceló, P., Kostylev, E. V., Monet, M., Pérez, J., Reutter, J., and Silva, J. P. The logical expressiveness of graph neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1lZ7AEKvB.

Batatia, I., Benner, P., Chiang, Y., Elena, A. M., Kovács, D. P., Riebesell, J., Advincula, X. R., Asta, M., Baldwin, W. J., Bernstein, N., Bhowmik, A., Blau, S. M., Cărare, V., Darby, J. P., De, S., Pia, F. D., Deringer, V. L., Elijošius, R., El-Machachi, Z., Fako, E., Ferrari, A. C., Genreith-Schriever, A., George, J., Goodall, R. E. A., Grey, C. P., Han, S., Handley, W., Heenen, H. H., Hermansson, K., Holm, C., Jaafar, J., Hofmann, S., Jakob, K. S., Jung, H., Kapil, V., Kaplan, A. D., Karimitari, N., Kroupa, N., Kullgren, J., Kuner, M. C., Kuryla, D., Liepuoniute, G., Margraf, J. T., Magdău, I.-B., Michaelides, A., Moore, J. H., Naik, A. A., Niblett, S. P., Norwood, S. W., O'Neill, N., Ortner, C., Persson, K. A., Reuter, K., Rosen, A. S., Schaaf, L. L., Schran, C., Sivonxay, E., Stenczel, T. K., Svahn, V., Sutton, C., van der Oord, C., Varga-Umbrich, E., Vegge, T., Vondrák, M., Wang, Y., Witt, W. C., Zills, F., and Csányi, G. A foundation model for atomistic materials chemistry, 2023.

Battiston, F., Cencetti, G., Iacopini, I., Latora, V., Lucas, M., Patania, A., Young, J.-G., and Petri, G. Networks beyond pairwise interactions: Structure and dynamics. *Physics Reports*, 874:1–92, 2020.

Beaini, D., Huang, S., Cunha, J. A., Moisescu-Pareja, G., Dymov, O., Maddrell-Mander, S., McLean, C., Wenkel, F., Müller, L., Mohamud, J. H., et al. Towards foundational models for molecular learning on large-scale multi-task datasets. *arXiv preprint arXiv:2310.04292*, 2023.

BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N., and Reddy, S. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*, 2024.

Benson, A. R., Gleich, D. F., and Leskovec, J. Higher-order organization of complex networks. *Science*, 353(6295): 163–166, 2016.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., and Ramesh, A. Video generation models as world simulators. 2024. URL https://openai.com/research/video-generation-models-as-world-simulators.

Brugere, I., Gallagher, B., and Berger-Wolf, T. Y. Network structure inference, a survey: Motivations, methods, and applications. *ACM Computing Surveys (CSUR)*, 51(2): 1–39, 2018.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Cai, C. and Wang, Y. A note on over-smoothing for graph neural networks. *arXiv preprint arXiv:2006.13318*, 2020.

Cao, Y., Xu, J., Yang, C., Wang, J., Zhang, Y., Wang, C., Chen, L., and Yang, Y. When to pre-train graph neural networks? an answer from data generation perspective! *arXiv preprint arXiv:2303.16458*, 2023.

Chai, Z., Zhang, T., Wu, L., Han, K., Hu, X., Huang, X., and Yang, Y. Graphllm: Boosting graph reasoning ability of large language model. *arXiv preprint arXiv:2310.05845*, 2023.

Chamberlain, B. P., Shirobokov, S., Rossi, E., Frasca, F., Markovich, T., Hammerla, N., Bronstein, M. M., and Hansmire, M. Graph neural networks for link prediction with subgraph sketching. *arXiv preprint arXiv:2209.15486*, 2022.

Chawla, N. V. and Karakoulas, G. Learning from labeled and unlabeled data: An empirical study across techniques and domains. *Journal of Artificial Intelligence Research*, 23:331–366, 2005.

Chen, D., Zhu, Y., Zhang, J., Du, Y., Li, Z., Liu, Q., Wu, S., and Wang, L. Uncovering neural scaling laws in molecular representation learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023a. URL https://openreview.net/forum?id=Ys8RmfF9w1.

Chen, N., Li, Y., Tang, J., and Li, J. Graphwiz: An instruction-following language model for graph problems. *arXiv preprint arXiv:2402.16029*, 2024a.

Chen, R., Zhao, T., Jaiswal, A., Shah, N., and Wang, Z. Llaga: Large language and graph assistant. *arXiv preprint arXiv:2402.08170*, 2024b.

Chen, Z., Liu, J., Wang, X., and Yin, W. On representing linear programs by graph neural networks. In *The Eleventh International Conference on Learning Representations*, 2022.

Chen, Z., Mao, H., Li, H., Jin, W., Wen, H., Wei, X., Wang, S., Yin, D., Fan, W., Liu, H., and Tang, J. Exploring the potential of large language models (llms) in learning on graphs. *ArXiv*, abs/2307.03393, 2023b.

Chien, E., Peng, J., Li, P., and Milenkovic, O. Adaptive universal generalized pagerank graph neural network. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=n6jl7fLxrP.

Dessí, D., Osborne, F., Recupero, D. R., Buscaldi, D., and Motta, E. Cs-kg: A large-scale knowledge graph of research entities and claims in computer science. In *International Workshop on the Semantic Web*, 2022. URL https://api.semanticscholar.org/CorpusID:253021556.

Di Giovanni, F., Rusch, T. K., Bronstein, M. M., Deac, A., Lackenby, M., Mishra, S., and Veličković, P. How does over-squashing affect the power of gnns? *arXiv preprint arXiv:2306.03589*, 2023.

Dong, K., Mao, H., Guo, Z., and Chawla, N. V. Universal link predictor by in-context learning. *arXiv preprint arXiv:2402.07738*, 2024.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Dong, Y., Johnson, R. A., Xu, J., and Chawla, N. V. Structural diversity and homophily: A study across more than one hundred big networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 807–816, 2017.

Dziri, N., Lu, X., Sclar, M., Li, X. L., Jian, L., Lin, B. Y., West, P., Bhagavatula, C., Bras, R. L., Hwang, J. D., et al. Faith and fate: Limits of transformers on compositionality. *arXiv preprint arXiv:2305.18654*, 2023.

Edmonds, J. and Johnson, E. L. Matching, euler tours and the chinese postman. *Mathematical Programming*, 5:88–124, 1973. URL https://api.semanticscholar.org/CorpusID:15249924.

Fatemi, B., Halcrow, J., and Perozzi, B. Talk like a graph: Encoding graphs for large language models. *arXiv preprint arXiv:2310.04560*, 2023.

Fey, M. and Lenssen, J. E. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

Fey, M., Lenssen, J. E., Weichert, F., and Leskovec, J. Gnnautoscale: Scalable and expressive graph neural networks via historical embeddings. In *International conference on machine learning*, pp. 3294–3304. PMLR, 2021.

Freitas, S., Duggal, R., and Chau, D. H. Malnet: A large-scale image database of malicious software. *arXiv preprint arXiv:2102.01072*, 2021.

Galkin, M., Yuan, X., Mostafa, H., Tang, J., and Zhu, Z. Towards foundation models for knowledge graph reasoning. *arXiv preprint arXiv:2310.04562*, 2023.

Galkin, M., Zhou, J., Ribeiro, B., Tang, J., and Zhu, Z. Zero-shot logical query reasoning on any knowledge graph. *arXiv preprint arXiv:2404.07198*, 2024.

Gao, J., Zhou, Y., and Ribeiro, B. Double permutation equivariance for knowledge graph completion. *arXiv preprint arXiv:2302.01313*, 2023a.

Gao, J., Zhou, Y., Zhou, J., and Ribeiro, B. Double equivariance for inductive link prediction for both new nodes and new relation types. In *NeurIPS 2023 Workshop: New Frontiers in Graph Learning*, 2023b.

Granovetter, M. S. The strength of weak ties. *American journal of sociology*, 78(6):1360–1380, 1973.

Gruver, N., Finzi, M., Qiu, S., and Wilson, A. G. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024a.

Gruver, N., Sriram, A., Madotto, A., Wilson, A. G., Zitnick, C. L., and Ulissi, Z. Fine-tuned language models generate stable inorganic materials as text. *arXiv preprint arXiv:2402.04379*, 2024b.

Gupta, S., Manchanda, S., Ranu, S., and Bedathur, S. J. Grafenne: learning on graphs with heterogeneous and dynamic feature sets. In *International Conference on Machine Learning*, pp. 12165–12181. PMLR, 2023.

Han, X., Zhang, Z., Ding, N., Gu, Y., Liu, X., Huo, Y., Qiu, J., Yao, Y., Zhang, A., Zhang, L., et al. Pre-trained models: Past, present and future. *AI Open*, 2:225–250, 2021.

Hassani, K. and Khasahmadi, A. H. Contrastive multi-view representation learning on graphs. In *Proceedings of International Conference on Machine Learning*, pp. 3451–3461. 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

He, X., Bresson, X., Laurent, T., Perold, A., LeCun, Y., and Hooi, B. Harnessing explanations: Llm-to-lm interpreter for enhanced text-attributed graph representation learning, 2023.

He, X., Tian, Y., Sun, Y., Chawla, N. V., Laurent, T., LeCun, Y., Bresson, X., and Hooi, B. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *arXiv preprint arXiv:2402.07630*, 2024.

Hibshman, J. I., Gonzalez, D., Sikdar, S., and Weninger, T. Joint subgraph-to-subgraph transitions: Generalizing triadic closure for powerful and interpretable graph modeling. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 815–823, 2021.

Hočevar, T. and Demšar, J. A combinatorial approach to graphlet counting. *Bioinformatics*, 30(4):559–565, 2014.

Hou, Z., Liu, X., Cen, Y., Dong, Y., Yang, H., Wang, C., and Tang, J. Graphmae: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 594–604, 2022.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*, 2019.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

Huang, H., Tang, J., Liu, L., Luo, J., and Fu, X. Triadic closure pattern analysis and prediction in social networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(12):3374–3389, 2015.

Huang, Q., Ren, H., Chen, P., Kržmanc, G., Zeng, D., Liang, P., and Leskovec, J. Prodigy: Enabling in-context learning over graphs. *arXiv preprint arXiv:2305.12600*, 2023a.

Huang, S., Poursafaei, F., Danovitch, J., Fey, M., Hu, W., Rossi, E., Leskovec, J., Bronstein, M., Rabusseau, G., and Rabbany, R. Temporal graph benchmark for machine learning on temporal graphs. *arXiv preprint arXiv:2307.01026*, 2023b.

Huang, X., Orth, M. R., Ceylan, İ. İ., and Barceló, P. A theory of link prediction via relational weisfeiler-leman. *arXiv preprint arXiv:2302.02209*, 2023c.

Huang, Y., Lu, W., Robinson, J., Yang, Y., Zhang, M., Jegelka, S., and Li, P. On the stability of expressive positional encodings for graph neural networks. *arXiv preprint arXiv:2310.02579*, 2023d.

Ibarz, B., Kurin, V., Papamakarios, G., Nikiforou, K., Bennani, M., Csordás, R., Dudzik, A. J., Bošnjak, M., Vitvitskyi, A., Rubanova, Y., et al. A generalist neural algorithmic learner. In *Learning on graphs conference*, pp. 2–1. PMLR, 2022.

Jeh, G. and Widom, J. Simrank: a measure of structural-context similarity. In *Proceedings of the eighth ACM*

SIGKDD international conference on Knowledge discovery and data mining, pp. 538–543, 2002.

Jin, B., Liu, G., Han, C., Jiang, M., Ji, H., and Han, J. Large language models on graphs: A comprehensive survey. *arXiv preprint arXiv:2312.02783*, 2023a.

Jin, B., Zhang, W., Zhang, Y., Meng, Y., Zhang, X., Zhu, Q., and Han, J. Patton: Language model pretraining on text-rich networks. *arXiv preprint arXiv:2305.12268*, 2023b.

Jin, W., Barzilay, R., and Jaakkola, T. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*, pp. 4839–4848. PMLR, 2020a.

Jin, W., Derr, T., Liu, H., Wang, Y., Wang, S., Liu, Z., and Tang, J. Self-supervised learning on graphs: Deep insights and new direction. *arXiv preprint arXiv:2006.10141*, 2020b.

Jing, Y., Yuan, C., Ju, L., Yang, Y., Wang, X., and Tao, D. Deep graph reprogramming. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24345–24354, 2023.

Jo, J., Lee, S., and Hwang, S. J. Score-based generative modeling of graphs via the system of stochastic differential equations. In *International Conference on Machine Learning*, pp. 10362–10383. PMLR, 2022.

Ju, M., Zhao, T., Wen, Q., Yu, W., Shah, N., Ye, Y., and Zhang, C. Multi-task self-supervised graph neural networks enable stronger task generalization. 2023.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Katz, L. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

Khanam, K. Z., Srivastava, G., and Mago, V. The homophily principle in social network analysis. *arXiv preprint arXiv:2008.10383*, 2020.

Kim, J., Nguyen, T. D., Min, S., Cho, S., Lee, M., Lee, H., and Hong, S. Pure transformers are powerful graph learners. *arXiv*, abs/2207.02505, 2022. URL https://arxiv.org/abs/2207.02505.

Kipf, T. N. and Welling, M. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

Klicpera, J., Bojchevski, A., and Günnemann, S. Predict then propagate: Graph neural networks meet personalized pagerank. In *International Conference on Learning Representations*, 2018.

Kreuzer, D., Beaini, D., Hamilton, W., Létourneau, V., and Tossou, P. Rethinking graph transformers with spectral attention. *Advances in Neural Information Processing Systems*, 34:21618–21629, 2021.

Kriege, N. M., Johansson, F. D., and Morris, C. A survey on graph kernels. *Applied Network Science*, 5(1):1–42, 2020.

Leskovec, J. and Krevl, A. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

Leskovec, J., Chakrabarti, D., Kleinberg, J., Faloutsos, C., and Ghahramani, Z. Kronecker graphs: an approach to modeling networks. *Journal of Machine Learning Research*, 11(2), 2010.

Li, J., Shomer, H., Ding, J., Wang, Y., Ma, Y., Shah, N., Tang, J., and Yin, D. Are graph neural networks really helpful for knowledge graph completion? *arXiv preprint arXiv:2205.10652*, 2022.

Li, J., Shomer, H., Mao, H., Zeng, S., Ma, Y., Shah, N., Tang, J., and Yin, D. Evaluating graph neural networks for link prediction: Current pitfalls and new benchmarking. *arXiv preprint arXiv:2306.10453*, 2023a.

Li, Y., Li, Z., Wang, P., Li, J., Sun, X., Cheng, H., and Yu, J. X. A survey of graph meets large language model: Progress and future directions. *arXiv preprint arXiv:2311.12399*, 2023b.

Li, Y., Xiong, M., and Hooi, B. Graphcleaner: Detecting mislabelled samples in popular graph learning benchmarks. *arXiv preprint arXiv:2306.00015*, 2023c.

Li, Y., Wang, P., Li, Z., Yu, J. X., and Li, J. Zerog: Investigating cross-dataset zero-shot transferability in graphs. *arXiv preprint arXiv:2402.11235*, 2024.

Lim, D., Hohne, F., Li, X., Huang, S. L., Gupta, V., Bhalerao, O., and Lim, S. N. Large scale learning on non-homophilous graphs: New benchmarks and strong simple methods. *Advances in Neural Information Processing Systems*, 34:20887–20902, 2021.

Lim, D., Robinson, J. D., Zhao, L., Smidt, T., Sra, S., Maron, H., and Jegelka, S. Sign and basis invariant networks for spectral graph representation learning. In *The Eleventh International Conference on Learning Representations*, 2022.

Ling, X., Wu, L., Wang, S., Pan, G., Ma, T., Xu, F., Liu, A. X., Wu, C., and Ji, S. Deep graph matching and searching for semantic code retrieval. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(5):1–21, 2021.

Liu, G., Inae, E., Zhao, T., Xu, J., Luo, T., and Jiang, M. Data-centric learning from unlabeled graphs with diffusion model. *arXiv preprint arXiv:2303.10108*, 2023a.

Liu, H., Feng, J., Kong, L., Liang, N., Tao, D., Chen, Y., and Zhang, M. One for all: Towards training one graph model for all classification tasks. *arXiv preprint arXiv:2310.00149*, 2023b.

Liu, H., Liao, N., and Luo, S. Simga: A simple and effective heterophilous graph neural network with efficient global aggregation. *arXiv preprint arXiv:2305.09958*, 2023c.

Liu, J., Yang, C., Lu, Z., Chen, J., Li, Y., Zhang, M., Bai, T., Fang, Y., Sun, L., Yu, P. S., et al. Towards graph foundation models: A survey and beyond. *arXiv preprint arXiv:2310.11829*, 2023d.

Liu, J., Mao, H., Chen, Z., Zhao, T., Shah, N., and Tang, J. Neural scaling laws on graphs. 2024a.

Liu, N., Wang, X., Bo, D., Shi, C., and Pei, J. Revisiting graph contrastive learning from the perspective of graph spectrum. *Advances in Neural Information Processing Systems*, 35:2972–2983, 2022.

Liu, R., Wang, Y., Xu, H., Liu, B., Sun, J., Guo, Z., and Ma, W. Source code vulnerability detection: Combining code language models and code property graphs. *arXiv preprint arXiv:2404.14719*, 2024b.

Liu, Z., Shi, Y., Zhang, A., Zhang, E., Kawaguchi, K., Wang, X., and Chua, T.-S. Rethinking tokenizer and decoder in masked graph modeling for molecules. In *NeurIPS*, 2023e. URL https://openreview.net/forum?id=fWLf8DV0fI.

Liu, Z., Yu, X., Fang, Y., and Zhang, X. Graphprompt: Unifying pre-training and downstream tasks for graph neural networks. In *Proceedings of the ACM Web Conference 2023*, pp. 417–428, 2023f.

Lu, S., Gao, Z., He, D., Zhang, L., and Ke, G. Highly accurate quantum chemical property prediction with unimol+. *arXiv preprint arXiv:2303.16982*, 2023.

Luan, S., Hua, C., Lu, Q., Zhu, J., Zhao, M., Zhang, S., Chang, X.-W., and Precup, D. Is heterophily a real nightmare for graph neural networks to do node classification? *arXiv preprint arXiv:2109.05641*, 2021.

Luan, S., Hua, C., Xu, M., Lu, Q., Zhu, J., Chang, X.-W., Fu, J., Leskovec, J., and Precup, D. When do graph neural networks help with node classification: Investigating the homophily principle on node distinguishability. *arXiv preprint arXiv:2304.14274*, 2023.

Luo, Y., Yan, K., and Ji, S. Graphdf: A discrete flow model for molecular graph generation. In *International Conference on Machine Learning*, pp. 7192–7203. PMLR, 2021.

Luo, Z., Song, X., Huang, H., Lian, J., Zhang, C., Jiang, J., Xie, X., and Jin, H. Graphinstruct: Empowering large language models with graph understanding and reasoning capability. *arXiv preprint arXiv:2403.04483*, 2024.

Ma, Y., Liu, X., Shah, N., and Tang, J. Is homophily a necessity for graph neural networks? *arXiv preprint arXiv:2106.06134*, 2021.

Mao, H., Chen, Z., Jin, W., Han, H., Ma, Y., Zhao, T., Shah, N., and Tang, J. Demystifying structural disparity in graph neural networks: Can one size fit all? *arXiv preprint arXiv:2306.01323*, 2023a.

Mao, H., Li, J., Shomer, H., Li, B., Fan, W., Ma, Y., Zhao, T., Shah, N., and Tang, J. Revisiting link prediction: A data perspective. *arXiv preprint arXiv:2310.00793*, 2023b.

Mao, H., Liu, G., Ma, Y., Wang, R., and Tang, J. A data generation perspective to the mechanism of in-context learning. *arXiv preprint arXiv:2402.02212*, 2024.

Masters, D., Dean, J., Klaser, K., Li, Z., Maddrell-Mander, S., Sanders, A., Helal, H., Beker, D., Rampášek, L., and Beaini, D. Gps++: An optimised hybrid mpnn/transformer for molecular property prediction. *arXiv preprint arXiv:2212.02229*, 2022.

McCoy, R. T., Yao, S., Friedman, D., Hardy, M., and Griffiths, T. L. Embers of autoregression: Understanding large language models through the problem they are trained to solve. *arXiv preprint arXiv:2309.13638*, 2023.

Menczer, F., Fortunato, S., and Davis, C. A. *A First Course in Network Science*. Cambridge University Press, 2020.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827, 2002.

Mishra, S., Panda, R., Phoo, C. P., Chen, C.-F. R., Karlinsky, L., Saenko, K., Saligrama, V., and Feris, R. S. Task2sim: Towards effective pre-training and transfer from synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9194–9204, 2022.

Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 4602–4609, 2019.

Morris, C., Kriege, N. M., Bause, F., Kersting, K., Mutzel, P., and Neumann, M. Tudataset: A collection of benchmark datasets for learning with graphs. In *ICML 2020 Workshop on Graph Representation Learning and Beyond (GRL+ 2020)*, 2020. URL www.graphlearning.io.

Morris, C., Lipman, Y., Maron, H., Rieck, B., Kriege, N. M., Grohe, M., Fey, M., and Borgwardt, K. Weisfeiler and leman go machine learning: The story so far. *Journal of Machine Learning Research*, 24(333):1–59, 2023. URL http://jmlr.org/papers/v24/22-0240.html.

Müller, L., Galkin, M., Morris, C., and Rampášek, L. Attending to graph transformers. *arXiv preprint arXiv:2302.04181*, 2023.

Murase, Y., Jo, H. H., Török, J., Kertész, J., Kaski, K., et al. Structural transition in social networks. 2019.

Oono, K. and Suzuki, T. Graph neural networks exponentially lose expressive power for node classification. In *International Conference on Learning Representations*, 2019.

Panwar, M., Ahuja, K., and Goyal, N. In-context learning through the bayesian prism. In *The Twelfth International Conference on Learning Representations*, 2023.

Perozzi, B., Fatemi, B., Zelle, D., Tsitsulin, A., Kazemi, M., Al-Rfou, R., and Halcrow, J. Let your graph do the talking: Encoding structured data for llms. *arXiv preprint arXiv:2402.05862*, 2024.

Project, U. C. R. Recommender systems and personalization datasets. URL https://cseweb.ucsd.edu/~jmcauley/datasets.html.

Prystawski, B. and Goodman, N. D. Why think step-by-step? reasoning emerges from the locality of experience. *arXiv preprint arXiv:2304.03843*, 2023.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Rafi, M. N., Kim, D. J., Chen, A. R., Chen, T.-H., and Wang, S. Towards better graph neural neural network-based fault localization through enhanced code representation. *arXiv preprint arXiv:2404.04496*, 2024.

Ramsundar, B., Eastman, P., Walters, P., Pande, V., Leswing, K., and Wu, Z. *Deep Learning for the Life Sciences*. O'Reilly Media, 2019. https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837.

Ribeiro, P., Silva, F., and Kaiser, M. Strategies for network motifs discovery. In *2009 Fifth IEEE International Conference on e-Science*, pp. 80–87. IEEE, 2009.

Ribeiro, P., Paredes, P., Silva, M. E., Aparicio, D., and Silva, F. A survey on subgraph counting: concepts, algorithms, and applications to network motifs and graphlets. *ACM Computing Surveys (CSUR)*, 54(2):1–36, 2021.

Robins, G., Pattison, P., Kalish, Y., and Lusher, D. An introduction to exponential random graph (p*) models for social networks. *Social networks*, 29(2):173–191, 2007.

Rossi, R. A. and Ahmed, N. K. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL http://networkrepository.com.

Ruiz, L., Chamon, L. F. O., and Ribeiro, A. Transferability properties of graph neural networks. *IEEE Transactions on Signal Processing*, 71:3474–3489, 2023. doi: 10.1109/TSP.2023.3297848.

Saparov, A. and He, H. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. *arXiv preprint arXiv:2210.01240*, 2022.

Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I., and Welling, M. Modeling relational data with graph convolutional networks. In *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*, pp. 593–607. Springer, 2018.

Shi, H., Ding, J., Cao, Y., Liu, L., Li, Y., et al. Learning symbolic models for graph-structured physical mechanism. In *The Eleventh International Conference on Learning Representations*, 2022.

Shoghi, N., Kolluru, A., Kitchin, J. R., Ulissi, Z. W., Zitnick, C. L., and Wood, B. M. From molecules to materials: Pre-training large generalizable models for atomic property prediction, 2023.

Srinivasan, B. and Ribeiro, B. On the equivalence between positional node embeddings and structural graph representations. *arXiv preprint arXiv:1910.00452*, 2019.

Stechly, K., Marquez, M., and Kambhampati, S. Gpt-4 doesn't know it's wrong: An analysis of iterative prompting for reasoning problems. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023.

Sun, F.-Y., Hoffmann, J., Verma, V., and Tang, J. Infograph: Unsupervised and semi-supervised graph-level

representation learning via mutual information maximization. *arXiv preprint arXiv:1908.01000*, 2019.

Sun, M., Zhou, K., He, X., Wang, Y., and Wang, X. Gppt: Graph pre-training and prompt tuning to generalize graph neural networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pp. 1717–1727, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393850. doi: 10.1145/3534678.3539249. URL https://doi.org/10.1145/3534678.3539249.

Sun, X., Cheng, H., Li, J., Liu, B., and Guan, J. All in one: Multi-task prompting for graph neural networks. 2023.

Taguchi, H., Liu, X., and Murata, T. Graph convolutional networks for graphs containing missing features. *Future Generation Computer Systems*, 117:155–168, 2021.

Tang, J. Aminer: Toward understanding big scholar data. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, WSDM '16, pp. 467, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450337168. doi: 10.1145/2835776.2835849. URL https://doi.org/10.1145/2835776.2835849.

Tang, J., Yang, Y., Wei, W., Shi, L., Su, L., Cheng, S., Yin, D., and Huang, C. Graphgpt: Graph instruction tuning for large language models. *arXiv preprint arXiv:2310.13023*, 2023.

Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E., Poulton, A., Kerkez, V., and Stojnic, R. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022.

Topping, J., Di Giovanni, F., Chamberlain, B. P., Dong, X., and Bronstein, M. M. Understanding over-squashing and bottlenecks on graphs via curvature. In *International Conference on Learning Representations*, 2021.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Trinh, T. H., Wu, Y., Le, Q. V., He, H., and Luong, T. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482, 2024.

Um, D., Park, J., Park, S., and young Choi, J. Confidence-based feature imputation for graphs with partially known features. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=YPKBIILy-Kt.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Vashishth, S., Sanyal, S., Nitin, V., and Talukdar, P. Composition-based multi-relational graph convolutional networks. In *International Conference on Learning Representations*, 2019.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

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

Veličković, P., Fedus, W., Hamilton, W. L., Liò, P., Bengio, Y., and Hjelm, R. D. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

Veličković, P., Badia, A. P., Budden, D., Pascanu, R., Banino, A., Dashevskiy, M., Hadsell, R., and Blundell, C. The clrs algorithmic reasoning benchmark. In *International Conference on Machine Learning*, pp. 22084–22102. PMLR, 2022.

Vignac, C., Krawczuk, I., Siraudin, A., Wang, B., Cevher, V., and Frossard, P. Digress: Discrete denoising diffusion for graph generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=UaAD-Nu86WX.

Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., and Borgwardt, K. M. Graph kernels. *Journal of Machine Learning Research*, 11:1201–1242, 2010.

Wang, H., Yin, H., Zhang, M., and Li, P. Equivariant and stable positional encoding for more powerful graph neural networks. In *International Conference on Learning Representations*, 2021.

Wang, H., Feng, S., He, T., Tan, Z., Han, X., and Tsvetkov, Y. Can language models solve graph problems in natural language? In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL https://openreview.net/forum?id=UDqHhbqYJV.

Wang, J., Guo, Y., Yang, L., and Wang, Y. Understanding heterophily for graph neural networks. *arXiv preprint arXiv:2401.09125*, 2024a.

Wang, J., Wu, J., Hou, Y., Liu, Y., Gao, M., and McAuley, J. Instructgraph: Boosting large language models via graph-centric instruction tuning and preference alignment. *arXiv preprint arXiv:2402.08785*, 2024b.

Wang, X., Yang, H., and Zhang, M. Neural common neighbor with completion for link prediction. *arXiv preprint arXiv:2302.00890*, 2023b.

Wang, Y., Elhag, A. A., Jaitly, N., Susskind, J. M., and Bautista, M. A. Generating molecular conformer fields. *arXiv preprint arXiv:2311.17932*, 2023c.

Wang, Y., Cui, H., and Kleinberg, J. Microstructures and accuracy of graph recall by large language models. *arXiv preprint arXiv:2402.11821*, 2024c.

Wu, Q., Zhao, W., Li, Z., Wipf, D. P., and Yan, J. Nodeformer: A scalable graph structure learning transformer for node classification. *Advances in Neural Information Processing Systems*, 35:27387–27401, 2022.

Wu, X., Ajorlou, A., Wu, Z., and Jadbabaie, A. Demystifying oversmoothing in attention-based graph neural networks. *arXiv preprint arXiv:2305.16102*, 2023.

Xia, J., Zhao, C., Hu, B., Gao, Z., Tan, C., Liu, Y., Li, S., and Li, S. Z. Mole-BERT: Rethinking pre-training graph neural networks for molecules. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=jevY-DtiZTR.

Xie, H., Zheng, D., Ma, J., Zhang, H., Ioannidis, V. N., Song, X., Ping, Q., Wang, S., Yang, C., Xu, Y., et al. Graph-aware language model pre-training on a large graph corpus can help multiple graph applications. *arXiv preprint arXiv:2306.02592*, 2023.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Xu, K., Li, J., Zhang, M., Du, S. S., Kawarabayashi, K.-i., and Jegelka, S. What can neural networks reason about? *ICLR 2020*, 2020.

Yang, L., Tian, Y., Xu, M., Liu, Z., Hong, S., Qu, W., Zhang, W., Cui, B., Zhang, M., and Leskovec, J. Vqgraph: Graph vector-quantization for bridging gnns and mlps. *arXiv preprint arXiv:2308.02117*, 2023.

Yang, Y., Liu, T., Wang, Y., Zhou, J., Gan, Q., Wei, Z., Zhang, Z., Huang, Z., and Wipf, D. Graph neural networks inspired by classical iterative algorithms. In *International Conference on Machine Learning*, pp. 11773–11783. PMLR, 2021.

Yao, Y., Wang, X., Zhang, Z., Qin, Y., Zhang, Z., Chu, X., Yang, Y., Zhu, W., and Mei, H. Exploring the potential of large language models in graph generation. *arXiv preprint arXiv:2403.14358*, 2024.

Yasunaga, M., Bosselut, A., Ren, H., Zhang, X., Manning, C. D., Liang, P., and Leskovec, J. Deep bidirectional language-knowledge graph pretraining. In *Neural Information Processing Systems (NeurIPS)*, 2022a.

Yasunaga, M., Leskovec, J., and Liang, P. Linkbert: Pretraining language models with document links. *arXiv preprint arXiv:2203.15827*, 2022b.

Ye, H., Zhang, N., Chen, H., and Chen, H. Generative knowledge graph construction: A review. *arXiv preprint arXiv:2210.12714*, 2022.

Ye, R., Zhang, C., Wang, R., Xu, S., and Zhang, Y. Language is all a graph needs. In *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 1955–1973, 2024.

Yin, H., Zhang, M., Wang, Y., Wang, J., and Li, P. Algorithm and system co-design for efficient subgraph-based graph representation learning. *arXiv preprint arXiv:2202.13538*, 2022.

Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., and Leskovec, J. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 974–983, 2018.

You, J., Ying, R., and Leskovec, J. Position-aware graph neural networks. In *International conference on machine learning*, pp. 7134–7143. PMLR, 2019.

You, J., Gomes-Selman, J. M., Ying, R., and Leskovec, J. Identity-aware graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 10737–10745, 2021.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., and Shen, Y. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

You, Y., Chen, T., Wang, Z., and Shen, Y. Graph domain adaptation via theory-grounded spectral regularization. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OysfLgrk8mk.

Yu, L., Lezama, J., Gundavarapu, N. B., Versari, L., Sohn, K., Minnen, D., Cheng, Y., Gupta, A., Gu, X., Hauptmann, A. G., et al. Language model beats diffusion–tokenizer is key to visual generation. *arXiv preprint arXiv:2310.05737*, 2023.

Yue, Z., Rabhi, S., Moreira, G. d. S. P., Wang, D., and Oldridge, E. Llamarec: Two-stage recommendation using large language models for ranking. *arXiv preprint arXiv:2311.02089*, 2023.

Zeng, H., Zhou, H., Srivastava, A., Kannan, R., and Prasanna, V. Graphsaint: Graph sampling based inductive learning method. *arXiv preprint arXiv:1907.04931*, 2019.

Zeng, H., Zhang, M., Xia, Y., Srivastava, A., Malevich, A., Kannan, R., Prasanna, V., Jin, L., and Chen, R. Decoupling the depth and scope of graph neural networks. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=d0MtHWY0NZ.

Zhai, G., Örnek, E. P., Wu, S.-C., Di, Y., Tombari, F., Navab, N., and Busam, B. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *arXiv preprint arXiv:2305.16283*, 2023.

Zhai, G., Örnek, E. P., Wu, S.-C., Di, Y., Tombari, F., Navab, N., and Busam, B. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *Advances in Neural Information Processing Systems*, 36, 2024.

Zhang, B., Gai, J., Du, Y., Ye, Q., He, D., and Wang, L. Beyond weisfeiler-lehman: A quantitative framework for gnn expressiveness. *arXiv preprint arXiv:2401.08514*, 2024.

Zhang, D., Liu, X., Zhang, X., Zhang, C., Cai, C., Bi, H., Du, Y., Qin, X., Huang, J., Li, B., Shan, Y., Zeng, J., Zhang, Y., Liu, S., Li, Y., Chang, J., Wang, X., Zhou, S., Liu, J., Luo, X., Wang, Z., Jiang, W., Wu, J., Yang, Y., Yang, J., Yang, M., Gong, F.-Q., Zhang, L., Shi, M., Dai, F.-Z., York, D. M., Liu, S., Zhu, T., Zhong, Z., Lv, J., Cheng, J., Jia, W., Chen, M., Ke, G., E, W., Zhang, L., and Wang, H. Dpa-2: Towards a universal large atomic model for molecular and material simulation. *arXiv preprint arXiv:2312.15492*, 2023a.

Zhang, F., Liu, X., Tang, J., Dong, Y., Yao, P., Zhang, J., Gu, X., Wang, Y., Kharlamov, E., Shao, B., Li, R., and Wang, K. Oag: Linking entities across large-scale heterogeneous knowledge graphs. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):9225–9239, 2023b. doi: 10.1109/TKDE.2022.3222168.

Zhang, M. and Chen, Y. Link prediction based on graph neural networks. *Advances in neural information processing systems*, 31, 2018.

Zhang, M., Li, P., Xia, Y., Wang, K., and Jin, L. Labeling trick: A theory of using graph neural networks for multi-node representation learning. *Advances in Neural Information Processing Systems*, 34:9061–9073, 2021.

Zhang, Y., Zhang, F., Yang, Z., and Wang, Z. What and how does in-context learning learn? bayesian model averaging, parameterization, and generalization. *arXiv preprint arXiv:2305.19420*, 2023c.

Zhang, Z., Li, H., Zhang, Z., Qin, Y., Wang, X., and Zhu, W. Graph meets llms: Towards large graph models. In *NeurIPS 2023 Workshop: New Frontiers in Graph Learning*, 2023d.

Zhang, Z., Luo, B., Lu, S., and He, B. Live graph lab: Towards open, dynamic and real transaction graphs with nft. *arXiv preprint arXiv:2310.11709*, 2023e.

Zhao, H., Liu, S., Chang, M., Xu, H., Fu, J., Deng, Z., Kong, L., and Liu, Q. Gimlet: A unified graph-text model for instruction-based molecule zero-shot learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Zhao, J., Zhuo, L., Shen, Y., Qu, M., Liu, K., Bronstein, M., Zhu, Z., and Tang, J. Graphtext: Graph reasoning in text space. *arXiv preprint arXiv:2310.01089*, 2023a.

Zhao, Q., Ren, W., Li, T., Xu, X., and Liu, H. Graphgpt: Graph learning with generative pre-trained transformers. *arXiv preprint arXiv:2401.00529*, 2023b.

Zheng, S., He, J., Liu, C., Shi, Y., Lu, Z., Feng, W., Ju, F., Wang, J., Zhu, J., Min, Y., Zhang, H., Tang, S., Hao, H., Jin, P., Chen, C., Noé, F., Liu, H., and Liu, T.-Y. Towards predicting equilibrium distributions for molecular systems with deep learning. *arXiv preprint arXiv:2306.05445*, 2023a.

Zheng, W., Huang, E. W., Rao, N., Wang, Z., and Subbian, K. You only transfer what you share: Intersection-induced graph transfer learning for link prediction. *arXiv preprint arXiv:2302.14189*, 2023b.

Zhong, Y., Shi, J., Yang, J., Xu, C., and Li, Y. Learning to generate scene graph from natural language supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1823–1834, 2021.

Zhu, Y., Xu, Y., Liu, Q., and Wu, S. An empirical study of graph contrastive learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021a.

Zhu, Z., Zhang, Z., Xhonneux, L.-P., and Tang, J. Neural bellman-ford networks: A general graph neural network framework for link prediction. *Advances in Neural Information Processing Systems*, 34:29476–29490, 2021b.

A. A Collection of Datasets to Support Pre-training

In this section, we show a collection of large-scale graph datasets from various fields to support pre-training massive-scale graph foundation models. We highly suggest using NetworkRepository (Rossi & Ahmed, 2015) for large-scale pretaining, which is the largest graph database presently available. Notably, burdensome pre-processing is required to clean those noisy and disordered data.

Table 1. A collection of datasets together with their URL and descriptions to support larger-scale pre-training

Name	URL	Description
TU-Dataset (Morris et al., 2020)	https://chrsmrrs.github.io/datasets/	A collection of graph-level prediction datasets
NetworkRepository (Rossi & Ahmed, 2015)	https://networkrepository.com/	The largest graph datasets, with graphs coming from 30+ different domains
Open Graph Benchmark (Hu et al., 2020)	https://ogb.stanford.edu/	Contains a bunch of large-scale graph benchmarks
Pyg (Fey & Lenssen, 2019)	https://pytorch-geometric.readthedocs.io	Official datasets provided by PYG, containing popular datasets for benchmark
SNAP (Leskovec & Krevl, 2014)	https://snap.stanford.edu/data/	Mainly focus on social network
Aminer (Tang, 2016)	https://www.aminer.cn/data/	A collection of academic graphs
OAG (Zhang et al., 2023b)	https://www.aminer.cn/open-academic-graph	A large-scale academic graph
MalNet (Freitas et al., 2021)	https://www.mal-net.org/#home	A large-scale function calling graph for malware detection
ScholKG (Dessí et al., 2022)	https://scholkg.kmi.open.ac.uk/	A large-scale scholarly knowledge graph
Graphium (Beaini et al., 2023)	https://github.com/datamol-io/graphium	A massive dataset for molecular property prediction
Live Graph Lab (Zhang et al., 2023e)	https://livegraphlab.github.io/	A large-scale temporal graph for NFT transactions
Temporal Graph Benchmark (Huang et al., 2023b)	https://docs.tgb.complexdatalab.com/	A large-scale benchmark for temporal graph learning
MoleculeNet (Ramsundar et al., 2019)	https://moleculenet.org/	A benchmark for molecular machine learning
Recsys data (Project)	https://cseweb.ucsd.edu/~jmcauley/datasets.html	A collection of datasets for recommender systems
LINKX (Lim et al., 2021)	https://github.com/CUAI/Non-Homophily-Large-Scale	A collection of large-scale non-homophilous graphs
CLRS (Veličković et al., 2022)	https://github.com/google-deepmind/clrs	A collection of algorithmic reasoning datasets.
GraphQA (He et al., 2024)	https://github.com/XiaoxinHe/G-Retriever	A collection of graph question answer datasets.

B. Existing GFMs

In this section, we demonstrate existing representative GFMs and categorize them into *primitive GFM*, *domain-specific GFM*, and *task-specific GFM*, as shown in Table 2.

Table 2. A collection of existing GFMs.

	Name	Domain	Task
Primitive GFM	PRODIGY (Huang et al., 2023a)	Text-attributed graph, Knowledge graph	Node classification, Knowledge graph reasoning
	OneForAll (Liu et al., 2023b)	Text-attributed graph, Knowledge graph, Molecule	Node classification, Knowledge graph reasoning, Graph classification
	LLaGA (Chen et al., 2024b)	Text-attributed graph	Node classification, Link Prediction, Graph classification
Domain-specific GFM	DIG (Zheng et al., 2023a)	Molecule	Molecular sampling, Property-guided structure generation.
	MACE-MP-0 (Batatia et al., 2023)	Material Science	Property predictions of solids, liquids, gases, and chemical reactions.
	JMP-1 (Shoghi et al., 2023)	Material Science	Atomic property prediction
	DPA-2 (Zhang et al., 2023a)	Material Science	Molecular simulation
	MoleBERT (Xia et al., 2023)	Molecule	Molecule property prediction
Task-specific GFM	ULTRA (Galkin et al., 2023)	Knowledge graph	Knowledge graph reasoning
	ULTRAQUERY (Galkin et al., 2024)	Knowledge graph	Knowledge graph reasoning
	Triplet-GMPNN (Ibarz et al., 2022)	General graph	algorithm reasoning
	G-Retriever (He et al., 2024)	General graph	Graph Question Answer
	GraphToken (Perozzi et al., 2024)	General graph	Graph Question Answer

C. Practical Recipes for GFM Applications

We primarily emphasize the graph principles revolving on transferability and neural scaling law in Section 3, and 4, respectively. In this section, we provide a comprehensive application discussion with more technical details. Specifically, we introduce the feature heterogeneity issue, pretext task design, and efficiency issues in subgraph-based methods in Appendix C.1, C.2, and C.3, respectively.

C.1. Tackling the Feature Heterogeneity Issue

Existing graph datasets cannot be uniformly utilized for pre-training due to the feature heterogeneity issue induced by missing features or different semantic spaces. Feature imputation techniques (Taguchi et al., 2021; Um et al., 2023; Gupta et al., 2023) are generally adapted to predict the missing attributes based on neighboring features. However, those techniques require each feature dimension to share the same semantic meaning. When features are from different semantic spaces, OFA (Liu et al., 2023b) manually converts the original features with text descriptions and then encodes the embedding with LLMs. Liu et al. (2023b) demonstrates the effectiveness and generality of using LLM embeddings to align heterogeneous node features in the text space. First, it shows that a large portion of feature heterogeneity is caused by the feature engineering process. For example, encoding text using Word2Vec (e.g. OGBN-Arxiv) and TF-IDF (e.g. Pubmed) results in different feature dimensions, leading to heterogeneity. If a unified LLM is utilized for encoding, feature heterogeneity can be solved. Second, for attributes without text attributes, OFA leverages multi-modal models to project them into textual descriptions. Specifically, OFA utilizes GIMLET (Zhao et al., 2024) to generate high-quality text descriptions for chemical molecules, and preliminarily shows that positive transferring can be achieved across diverse domains like text-attributed graphs, knowledge graphs, and molecule graphs after projecting heterogeneous features into text space. However, LLM embeddings still have limitations and their performance highly depends on the prompts provided to the LLM text encoder, remaining ample room for exploration in this area. One potential way is to borrow ideas from the CV domain. Yu et al. (2023) shows that after using LLMs to unify the feature space, further using discrete tokenization for the image can create a better latent space to further improve performance.

Feature misalignment can also be found in the inference stage between the pre-training model input and the test data. Jing et al. (2023) concatenates a learnable padding feature on the downstream task feature to align with the pre-trained GNN. However, such a technique cannot adapt to the case when the feature space is not aligned. Zhao et al. (2023a) directly abandons the original feature and utilizes the feature similarity as guidance.

C.2. Pretext Task Design

Given the scarcity of labeled data, a pretext task that can effectively utilize unsupervised data is the cornerstone for larger-scale neural scaling. We provide a brief review of the representative pretext designs.

Graph contrastive learning designs the pretext tasks (Sun et al., 2019; Veličković et al., 2018; Hassani & Khasahmadi, 2020; You et al., 2020) to obtain the equivalence via contrasting original and augmented views of the graph without materially changing the semantic content of the input. An initial unified understanding (Liu et al., 2022) on those pre-text tasks illustrates that existing pretext tasks focus on preserving the invariance with the low frequency on the graph spectrum. Nonetheless, different pretext tasks remain different where Zhu et al. (2021a) observes that satisfactory performance requires pretext tasks and downstream tasks share similar philosophies, such as homophily. To obtain a pre-training model that benefits different downstream tasks, Ju et al. (2023) adaptively combined pretext tasks with different philosophies via a multi-task learning framework.

The generative self-supervised learning designs the pretext tasks (Hou et al., 2022; Hu et al., 2019; Kipf & Welling, 2016) to capture the shared data generation process among different tasks. Particularly, they attempt to predict the masked portions of the graph using the remaining structure and features. Liu et al. (2023e); Xia et al. (2023) further observe that task granularity also plays an important role in generative modeling. Specifically, employing node-level pretext tasks may lead the model to learn only low-level features (Liu et al., 2023e) while ignoring the global information essential for graph-level tasks. To address this issue, they adopt a GNN-based tokenizer to explicitly model high-level information in the pre-training stage and thus improve the downstream task performance.

More recently, the next token prediction (NTP) pretext task (Zhao et al., 2023b) achieves initial success in the molecular graph. Notably, this is the first pretext task demonstrating empirical evidence of model scaling. The potential reason for its success may be (1) the construction of a fixed token set, narrowing down the problem space in a finite set to only predict a discrete token and (2) choosing transformers as the backbone model. However, it remains unclear whether the success can be easily extended to more tasks.

C.3. Efficiency Issues in Subgraph-based Methods.

Subgraph-based extraction is a widely adopted technique in GFM to achieve inductive inference (Zeng et al., 2021) and unify different task formulations (Sun et al., 2023; Liu et al., 2023b). Nonetheless, the subgraph-based extraction leads to

the following issues: (1) information loss in high-order neighborhoods; (2) duplicate sub-graph information with excessive memory consumption; (3) the time complexity of vanilla subgraph extraction grows exponentially with the number of hops, and (4) the online sub-graph sampling on the fly is also in non-acceptable inference latency (Yin et al., 2022).

Moreover, the subgraph-based method will increase the number of forward processes for a link-level task, leading to limited efficiency. Typically, for each node pair, we will extract a sub-graph based on them, and apply the forward process. Therefore, the number of forwards increases from $O(|N|)$ to $O(|E|)$, where $|N|$ and $|E|$ are the number of nodes and the number of edges. In practice, the subgraph-based method like SEAL[1] cannot be directly applied to the larger OGB-graph due to such efficiency issues. Those issues hinder the applicability of subgraph-based methods.

Designing an effective and efficient sampling method remains a major challenge in building GFM. Graph sampling techniques like (Zeng et al., 2019) and global state vectors (Fey et al., 2021) can help to alleviate these issues.

1. Existing GFM such as PRODIGY (Huang et al., 2023a) and OneForAll (Liu et al., 2023b) based on a subgraph-based view suffers from severe efficiency issues, especially on the link-level tasks. (1) For each node pair, those methods will extract a subgraph based on them, and apply the forward process. Therefore, the number of forwards increases from $O(|N|)$ to $O(|E|)$, where $|N|$ and $|E|$ are the number of nodes and the number of edges. (2) Moreover, the sampling subgraph may also introduce an efficiency problem (Yin et al., 2022). Subgraph-based methods sample subgraphs in either an offline or online manner. For offline sampling, they need to store subgraph patches for all possible queries, which introduces enormous memory overhead for large graphs. For online sampling, it samples subgraphs on the fly and results in non-acceptable inference latency.
2. To solve these efficiency issues, a potential approach is to convert GNN computing to feature precomputation (Chamberlain et al., 2022). This works for node-level and link-level tasks, but extending it to graph-level tasks is still challenging.

D. Additional Principles

D.1. Principles on deeper GNN design

In section 3, we emphasize principles revolving on the transferability across datasets. Besides, another line of principles focuses on tackling the model limitation towards building effectiveness deeper GNN to capture higher-order structural information.

Principles can tell why vanilla GNNs suffer from performance degradation when increasing the number of layers and provide guidance for solutions. The principles can be majorly categorized into the following three perspectives: (i) The over-squashing problem (Topping et al., 2021) illustrates that the node representation is insensitive to information from important but distant nodes. (ii) The over-smoothing problem (Oono & Suzuki, 2019; Cai & Wang, 2020) illustrates that more aggregations lead to the node representations converging to a unique equilibrium, which loses the distinction between different nodes. (iii) The underreaching (Barceló et al., 2020) illustrates the failure to explore, cover, or affect all relevant nodes in the graph, leading to information loss. Various techniques are proposed to identify the root causes (Di Giovanni et al., 2023; Wu et al., 2023) and solve the expressiveness issues via new GNN (Yang et al., 2021) and graph transformer (Wu et al., 2022; Müller et al., 2023) architecture designs.

Despite those principles are well-studied, they can have a different position and challenges when moving from end-to-end training GNNs to the GFM requiring models to apply across different tasks and datasets. Instead of only emphasizing the effectiveness on a single dataset, building GFM raises a novel challenge for us to get good performance with a unified model backbone on diverse datasets. The GNN backbone should be able to simultaneously capture discriminative low-order neighborhood information for homophily graphs and high-order neighborhood information for heterophily graphs while the low-order ones may be noisy. Current GFMs like OneForAll (Liu et al., 2023b) empirical solve such a problem via adding virtual nodes with proper prompt designs. Nonetheless, there remains a gap between building effective and adaptive deeper GNN for GFM and the current theoretical principles.

D.2. Additional Description on the Relational Graph Vocabulary of ULTRA

Typically, the relational vocabulary of ULTRA (Galkin et al., 2023) is inspired by the graph expressiveness theory in (Gao et al., 2023b). A concrete example of the relation representation can be found in Figure 2(a) in (Galkin et al., 2023). The relation vocabulary will provide the same embedding for the following two subgraphs with the same relational structure.

Michael Jackson $\xrightarrow{authored}$ *Thriller* $\xrightarrow{genre}$ *disco* seamlessly transfers to new entities *Beatles* $\xrightarrow{authored}$ *Let It Be* $\xrightarrow{genre}$ *rock* at inference time. They have the same relational structure with invariant representations regardless of permutations on different node types. Interpreting with the graph vocabulary perspective, those two subgraphs should be mapped into the same token.

Whether the relational vocabulary is suitable or not is according to the graph expressiveness theory. Typically, if two subgraphs are invariant with the same relational structure, i.e., isomorphic to the node type permutation, they will be mapping into the same token with the same representation. In contrast, if two nodes are not invariant, i.e., non-isomorphic to the node type permutation, they will be mapped into different tokens with different representations. Overall, the criterion for relational vocabulary is that two sub-structures can be mapped into the same token if and only if two sub-structures are isomorphic.

E. Discussions & Open questions

E.1. More discussions on LLMs and Graphs

In this section, we provide an extended discussion on leveraging LLMs for graph-related tasks, building on the concepts introduced in Section 4.4. We provide a more comprehensive discussion on the interaction between LLMs and Graphs.

Specifically, the current integration of graph and foundational models follows two primary pathways. The first involves using graphs to augment the capabilities of other foundational models. The second employs foundational models to address challenges encountered in graph machine learning. The first type of work focuses on enhancing the capabilities of foundation models by graphs. Yasunaga et al. (2022a;b); Jin et al. (2023b); Xie et al. (2023) further pre-train LLMs on text-attributed graphs with a structure-aware pretext task. For example, Yasunaga et al. (2022b) trains an LLM to predict masked edges, which is formalized as pair classification on two end nodes' attributes. Structure-aware training can effectively enhance language models' capability on those tasks requiring structure reasoning, such as multi-hop reasoning. These works still view the graph as a second-class citizen providing auxiliary information and put more emphasis on text-centric tasks like question answering (Yasunaga et al., 2022a;b).

The second line of work adopts LLMs' capabilities to solve challenges in the graph domain. Luo et al. (2024); Chen et al. (2024a); Wang et al. (2024b); Li et al. (2024); Ye et al. (2024) utilize the instruction fine-tuning the LLM for various capabilities including zero-shot (Li et al., 2024), link prediction (Ye et al., 2024), graph reasoning (Luo et al., 2024; Chen et al., 2024a; Wang et al., 2024b). Surprisingly, Wang et al. (2024b) observes that graph fine-tuning can even help those tasks irrelevant with graphs, e.g., mitigate hallucination, logic reasoning, and question answering. LLMs can also be utilized for graph generation (Yao et al., 2024; Wang et al., 2024c) where Wang et al. (2024c) finds that the graph generated by LLMs is biased towards more triangles and alternating 2-paths, leading to worse performance on the graph recall task.

Potential drawback on GNN-enhenced LLM Although these models can perform well, they still have two shortcomings: (1) The ability to process structures is bounded by the capabilities of GNN; (2) The instruction tuning can be costly while the tuned model can only tackle the corresponding downstream task and is not transferable to other tasks and datasets, which makes their capabilities distant from a GFM. We agree that LLM illustrates superior performance on textual node feature understanding. Nonetheless, it remains unclear whether LLM should play a key role in building GFM or just serve as a better textual feature encoder. Moreover, Stechly et al. (2023) observes that LLMs are bad at solving graph coloring instances even with multiple-round prompt. Yue et al. (2023) points out the efficiency issue of utilizing LLM on the recommendation, the downstream link prediction task. The effectiveness and efficiency of LLMs remains unclear.

E.2. Whether There exists a General Graph Vocabulary?

A shared graph vocabulary that is effective in transferability across domains and tasks remains an open question. In the current stage, we do not speculate the most ideal form of such vocabulary both across tasks and domains. Instead of transferring both across tasks and domains, the current vocabulary design can either transfer across tasks or domains. There is no unified graph vocabulary design at the current research state, as most of the graph vocabulary is either task or domain-specific, e.g., relational vocabulary in ULTRA (Galkin et al., 2023).

Despite the general graph vocabulary is challenging and not yet realized, we want to introduce one potential way toward it via graph tokenizer training, which is proposed in VQGraph (Yang et al., 2023). Specifically, it tokenizes nodes with similar structural properties into discrete codes using variants of VQ-VAE (Van Den Oord et al., 2017). After pre-training the tokenizer with a graph reconstruction objective, the discrete codes contained in the codebooks can represent typical structural

patterns. The properties of the learned codes will be based on two factors: (1) the encoder and decoder architecture; and (2) the pre-training objective. The current design is still under the guidance of graph principles, remaining not generalized across all the graphs. We leave the open question whether there is a universal structure space on graph as the future work

Is it possible for GFM to transfer across different domains? For instance, can a model trained on molecular data positively transfer to KG data? The answer to this question is initially yes, where we utilize the OneForAll model (Liu et al., 2023b) as a successful showcase on cross-domain transferring. The OneForAll model unifies feature spaces from different domains by using LLM embeddings, map features, and labels into a unified text space with better transferability. Such a unified space thereby serves as the basis for GFM that can transfer across citation networks, Wikipedia knowledge graphs, and molecular graphs. In the zero-shot setting, OneForAll shows that models trained on citation networks with the node classification task can show positive transfer on molecular graphs, even surpassing the performance on foundation models specific for the science domain like Galactica (Taylor et al., 2022). We hypothesize the potential reason is the existence of transferrable patterns among domains, e.g., shared motifs. Such transferrable patterns could be modeled by the ability to recognize cycles. For example, 6-cycles are seen in molecules while 3-cycles (triangles) are critical for social networks (Granovetter, 1973). Moreover, (Ribeiro et al., 2009) indicates that there are shared patterns between the electronic circuit, the transcriptional network, and the social network despite a severe domain shift. More investigations are needed to verify whether models can effectively utilize those transferrable patterns. For the transferability between knowledge graphs and molecular graphs, we do not have empirical evidence so far. We hypothesize that if the knowledge graph involves chemistry-related knowledge, positive transferability can be achievable.

E.3. Deeper GNNs as the Backbone of GFMs

The development of deep learning generally believes the benefit from deeper Neural Network (He et al., 2016), where the worst case of deeper Neural Networks should be degraded to a shallow solution. Notably, we want to emphasize the difference between the general deeper Neural Network design and the deeper GNN design. In general deeper Neural Network design, e.g., ResNet (He et al., 2016), Transformer (Vaswani et al., 2017), a deeper Neural Network naturally leads to larger parameter scaling. However, a deeper GNN does not necessarily lead to larger parameter scaling. The key reason is that the GNN is composed of two different components including (1) the feature transformation layer and (2) the aggregation layer. Many deeper GNNs focus on increasing the non-parametric aggregation layer while the number of feature transformation layers remains small. For instance, the APPNP (Klicpera et al., 2018) on Planetoid datasets generally only utilizes two feature transformation layers with a number of parameters less than 10,000. It remains skeptical whether deeper GNNs with careful aggregation function design can achieve similar success in other domains without scaling parameter size.

E.4. Is Invariance a Necessarity for Building GFM?

We propose a graph vocabulary perspective emphasizing the invariance among graphs is essential for building Graph Foundation Model. However, it remains a mystery whether we should implicitly keep such invariance via specific Message Passing Neural Network (MPNN) design with equivariance. On the one hand, (Galkin et al., 2023; 2024) indicates the effectiveness of building GFM with equivariance. On the other hand, (Abramson et al., 2024; Wang et al., 2023c) finds that it is unnesserary to ensure invariance or equivariance with respect to global rotations and translation of the molecule. Instead, data argumentation with random rotating and translating is utilized as an implicit regularization during training. (Perozzi et al., 2024), which first encodes graph with GNNs to conduct prompts and then utilizes LLM for prediction, observes that better performance when breaking the necessary equivariance. So far there is no agree on how to preserve geometric equivariance while LLMs also demonstrate potential . In additional to geometric Neural Network design, data augmentation, loss functions, and the potential expressiveness of LLM may also provide effective solution.

E.5. Comparison with Past Relevant Literature.

Concurrent to our position paper, Jin et al. (2023a); Li et al. (2023b); Zhang et al. (2023d) reviews those methods adapting large language model (LLM) for graph, which haven't shown transferring capabilities and thus diverge from our scope to build a graph-centric GFM. Liu et al. (2023d) further discusses existing graph pre-training and adaption techniques with a focus on their implementations. Instead of technical details, our work focuses more on the fundamental principles, e.g., geometric invariance across datasets. With principle guidance, we depict the promising and relatively elusive directions for the development of GFMs.