Graph Neural Network and Its Relation with WL Technique

Dr, Preeti Panwar[1]
[1]Associate Professor in Mathematics, Guru Nanak Khalsa College, Karnal

***Abstract*—The research revolved around the theoretical rationale and investigation of the WL algorithm and graph neural network. The study's two main objectives are to analyze the expressive power of graph neural networks and explore if they may outperform higher order WL algorithms. To effectively achieve the paper's goal, the study used non-experimental approach layered on top of a quantitative methodology. Along with searching for all the notable variants that are currently garnering attention from the perspective of theoretical analysis, the research also looks into the use of higher-order GNN and WL tests. Examining GNN's strength in conjunction with WL tests was the first step in the study process. From there, greater-order GNN and k-dimensional WL tests were investigated. Furthermore, the study emphasizes the challenges and limitations associated with conventional GNN and WL approaches, which may be addressed by applying more advanced GNN and WL techniques. Furthermore, when feature embedding is included, the expressive capability of GNNs outperforms that of the 1-WL test. However, more research is advised to fully examine the expressive capability of GNN and its higher variations.**

***Index Terms*—Graph Neural Network, WL Algorithm, Expressiveness of GNN, Higher-order GNN and WL technique.**

I.INTRODUCTION

Graph neural network, a type of supervised neural network particularly suited for applications involving graphs and nodes. A particular kind of neural network that works with graph-structured data is called a graph neural network (GNN). An immediate application of deep learning techniques to graphs is the GNN class. They come in assistance for analysis of graph activities including graph clustering, link prediction, and node categorization (Kumar, A et al., 2023; Li & Leskovec, 2022). According to Nunes and Pappa (2020), it is a GNN augmentation that preserves the characteristics of the recursive and random walk models. Given its ability to compute all types of graphs, including cyclic, directed, and undirected ones, the GNN represents a more sophisticated version of the recursive neural network. Additionally, it is capable of managing specific applications without requiring an initial processing phase (Chen, Z. et al., 2020). GNNs are an effective technique for resolving a variety of NLP issues. By introducing learning algorithms and increasing the types of computation that may be structured, the graph neural network further expanded the random walk theory.

Graph neural networks are also reliant on the information dissemination approach, which uses a collection of variables that are part of the graph's nodes and are computerized by the graph according to how connectedly they contain data. One of GNN's features is its expressiveness in identifying isomorphisms and categorizing graph variations that are investigated in the study using the MPNN framework (Bouritsas et al., 2022; Joshi et al., 2023). Tasks such as text categorization, relation extraction, user geolocation, and machine translation utilizing semantics have all been addressed by them. It is possible to train GNNs to anticipate tasks involving graphs, edges, and nodes. Nodes, which stand for entities (such as individuals, merchandise, or chemicals), and edges, which show the interactions or connections between the nodes, make up graphs. GNNs are very effective in relational data tasks like molecular chemistry, social network analysis, and recommendation systems (Hang, M et al., 2021; Maron, H et al., 2019). Their applications include text and picture classification, graph clustering and generation, link predictions, and graph and node classification.

Notwithstanding a number of characteristic graphs, neural networks face a number of difficulties in handling particular components, including the "architecture component, attention function, aggregation function, activation function, and hyperparameters," which are in charge of their dropout and learning rate (Guan, C. et al., 2022). These difficulties include space and time complexity issues.

A. Weisfeiler-Lehman (WL) algorithm

To ascertain if two graphs are fundamentally the same, one graph isomorphism approach is the Weisfeiler-Lehman (WL) algorithm. Since its introduction by B. Weisfeiler and A. Lehman in 1968, it has grown to be an essential tool in computer science and graph theory (Wang, Q et al., 2023). The WL method is an effective and user-friendly tool for assessing graph isomorphisms. It is useful in many domains, such as network analysis, chemistry, and computer science. Notwithstanding its efficacy, it is crucial to remember that the WL method is not a polynomial-time algorithm, and research on its complexity is currently ongoing.

Initially, labeling of the nodes accomplishes in a manner such as giving each node in the two graphs a distinct label. The degrees of the nodes—that is, the variety of edges adjacent to every node—are the basis for the labels at first. Followed by the level of iterations where every iteration combines the labeling of every node with the neighboring nodes' sorted list. Assign the node's appropriate identifier to this combined text (Bianchi & Lachi, 2023). Continue doing this until there are no more modifications, or for a predetermined number of repetitions. The graphs could be isomorphic if the last node identifiers in both graphs are the exact same. They don't qualify as isomorphic if they aren't (Maron, H et al., 2019).

The WL technique is very strong since it can identify isomorphism when other techniques are unable to. On the other hand, non-isomorphic graphs may sometimes end up with the same labels, and it may not always be possible to be able to identify them Morris, C et al., (2019; 2021). The WL method is a useful instrument in the area of graph theory along with associated subjects because of its efficiency and ability to be applied to different kinds of graphs. Graph isomorphism testing employs sophisticated algorithms and heuristics that are built upon this foundation.

Comparison Between GNN And WL based on their functionality, application, learning and testing, and complexities-

- GNNs: Graph-network architectures (GNNs) are a class of neural network architectures intended for graph-structured data learning and representation. They are employed in a number of different tasks, including graph classification, link prediction, and node classification.
- WL method: In contrast, the main goal of the WL method is to ascertain if two graphs are fundamentally isomorphic or identical.
- GNNs: Through the aggregate of data from nearby nodes, GNNs are able to learn node representations. They are adaptable for tasks requiring graph-structured data and are capable of capturing intricate relationships within a graph.
- WL Algorithm: Using neighborhood structures in the immediate area, the WL algorithm progressively improves node labels. It can ascertain if two graphs are fundamentally the same and is particularly intended for graph isomorphism testing.
- GNNs: GNNs are used in many different fields, such as recommendation systems, biology, chemistry, social network analysis, and more. They are employed in learning activities involving graphs where comprehension of linkages is essential.
- WL method: Testing graph isomorphisms is the main application of the WL method. It is an effective method for figuring out whether the structures of two graphs are equal.
- GNNs: Learning from data is the main goal of GNNs. In order to produce predictions or classifications on fresh, unseen graphs, they are trained on labeled graphs to identify patterns and correlations.
- WL Algorithm: This algorithm is used for testing and verification. It is used to compare the architecture of two supplied graphs; it does not learn from data.
- GNNs: Generic neural networks (GNNs) are intricate neural network structures that may include several layers, distinct aggregation functions, and attention processes. They can

identify complex patterns in big, complicated graphs.

- WL Algorithm: The WL algorithm uses node labels to perform iterative operations. Though conceptually strong, it is effective for some isomorphism testing tasks due to its simplicity.
- The main intention of the work is to explore the basic concept related with GNN and WL algorithms and their potency. The purpose of the paper is to render quantitative research an acceptable and desired approach. Because it is predicated on secondary technique, which gathers or draws from previously conducted research to advance knowledge, condense previously published resources, and ensure their efficacy in this study. The main goal of the research is to ascertain if graph neural networks along with its expressive power has capability when compared to the WL method. This determination is based on the majority of advancements in the field of research, including higher order GNN and higher order WL.

B. Contribution of this research-

- To provide clear vision regarding whether graph neural networks are more potent than higher order WL algorithms and the expressive potential of graph neural networks.
- The usefulness of higher-order GNN and WL tests, as well as looking for all the noteworthy variations that are now gaining attention from the standpoint of theoretical analysis.

C. Outline of this Research

The study briefly reviews notable and important previous studies that are pertinent to the manuscript's topic and are effectively merged in literature review. Research methodology is the portion that comes next, where the best technique is chosen to achieve the research's goal and fulfill the objectives in an easy-to-understand way. Subsequently, the data analysis part shed insight on the research's aim, which involved investigating the theoretical architecture of graph neural networks, WL algorithm their higher order and relevance with expressiveness. Eventually, the result and conclusion segment try to present the summaries outlook of the complete paper.

II. LITERATURE REVIEW

For many applications involving graphs, GNN are useful machine learning models. Many research endeavors center on the theoretical constraints of GNNs, namely their expressive capability, despite their shown performance in the real world. Scholarly studies in this field have mostly examined GNNs' capacity to recognize graph isomorphisms. More recent research (Zhang, B et al., 2023) has attempted to quantify the expressive capability of GNNs by utilizing features like subgraph counting and connection learning, which are more applicable and grounded in reality. Nevertheless, there is a lack of thorough summaries and open-source repositories that explore models in this crucial direction among survey publications. In order to close the gap, the paper did an initial survey to find models for improving expressive power under various definitional schemes.

Researchers (You, J et al., 2021) create a type of message-passing GNNs that surpasses the 1-WL test in expressive power: Identity-aware Graph Neural Networks (ID-GNNs). The constraints of current GNNs are addressed by ID-GNN, which provides a simple yet effective workaround. By taking into account nodes' identities inductively during message passing, ID-GNN expands on current GNN topologies. IDGNN initially retrieves a self-organizing system centered at a specific node, and then it performs cycles of heterogeneous message passing, where various sets of variables are applied to the center node in comparison to other neighboring sites in the ego network, in order to embed that node. The paper furthermore suggests a more straightforward but quicker variant of ID-GNN, which incorporates node identification data as enhanced node characteristics. Additionally, suggest a quicker yet more straightforward variant of ID-GNN that incorporates node identification data as enhanced node characteristics. The combined effect of both ID-GNN versions is a broadening expansion of message passing GNNs. Investigations indicate that converting pre-existing GNNs to ID-GNNs results in a typical enhancement of the precision of 40% on difficult node, edge, and graph property estimation assignments; 3% on node and graph segmentation standards; and 15% on ROC AUC on practical link estimation assignments (Ni, J et al., 2021).

One well-liked method for forecasting graph structured data is the use of graph neural networks, or GNNs. Since GNNs firmly interweave the input structure into the topology of the neural network, standard explainable AI techniques are not relevant. Until now, GNNs have mostly stayed unknown to the user. This research (Schnake, T et al., 2021) demonstrates that, in fact, GNNs may be naturally described by grouping edges that collectively contribute to the prediction, or by applying higher-order extensions. Applying established methods like layer-wise relevance propagation (LRP) at each stage, the paper discovers that such explanations may be practically derived through the use of a nested attribution scheme. The input graph's walks that are pertinent to the prediction are gathered into the output. The novel explanation method, which study denote by GNN-LRP, is applicable to a broad range of graph neural networks and lets us extract practically relevant insights on sentiment analysis of text data, structure-property relationships in quantum chemistry, and image classification.

The WL graph isomorphic behavior test has been demonstrated to restrict the expressive capability of typical GNNs, and it is from this evaluation that they acquire established constraints like the incapacity of recognizing and counting graph substructures. Nonetheless, there is strong empirical support—for instance, from the fields of genomics and network science—that structural components and their downstream activities are frequently closely linked. The proposal (Bouritsas, G et al., 2022), "Graph Substructure Networks" (GSN), is a message transmission system that considers topology and relies on substructure embedding. The analysis of the architecture's expressive potential theoretically demonstrates that it surpasses the WL test in terms of expressiveness, and we offer enough prerequisites for its universality. Crucially, it makes no effort to follow the WL hierarchy. By doing so, we are able to distinguish even challenging cases of graph isomorphism while preserving a number of desirable characteristics of conventional GNNs, including locality and linear complexity of networks.

The accepted method for learning with graph-structured data is now GNNs its potential as well as its limitations have been highlighted by earlier research. Regretfully, it was demonstrated that the expressiveness of ordinary GNNs is constrained. When it comes to differentiating non-isomorphic networks, these representations are no more effective than the 1-dimensional Weisfeiler-Leman (1-WL) technique. The paper (Michel, G et al., 2023) provides a new model in this study called Path Neural Networks (PathNNs), which aggregates pathways coming from nodes to update node representations. Three distinct iterations of the PathNN model are derived, each of which aggregates all shortest pathways, any straightforward routes of length up to K, and single shortest paths. Two of these variations are shown to be strictly more effective than the 1-WL algorithm, and theoretical findings are confirmed. The highest expressive PathNN variation is capable of discriminating among 3-WL indistinguishable graphs. PathNNs can identify couples of non-isomorphic vertices that are unidentifiable by 1-WL. Additionally, the various PathNN variations are tested on samples for classifying graphs and graph regression analysis, and they often beat the initial techniques.

It is known that the 1-WL technique upper-bends the expressiveness of MP-GNNs.Current efforts to create more potent GNNs either necessitate the use of ad hoc characteristics or entail highly complicated time and space operations. The study (Liu, M et al., 2022) presents a broad and provably strong GNN framework in this work that maintains the message forwarding scheme's scalability. Specifically, it first suggests incorporating edges between neighbors when empowering 1-WL for graph isomorphic relationships tests, resulting in NC-1-WL. It is demonstrated that, theoretically, the expressive ability of NC-1-WL is precisely beyond 1-WL and beneath 3-WL. Moreover, the paper provides the NC-GNN paradigm as a neural variant of NC-1-WL that is differentiable. It is demonstrably as effective to use this straightforward NC-GNN implementation as NC-1-WL.

While the research (Grohe, M. 2021) attempts to address a number of issues related to graph neural networks and their relationship to graph isomorphism, a number of issues remain unanswered, including complexity theoretic analysis, Boolean queries, unary queries, and other issues. The current GNN framework is based on the operation of neighboring node aggregation phenomena, which limits the capacity to discriminate, much as the one-dimensional WL isomorphic test. A novel method based on graphs convolutional functioning was described in the research (Damke, C. et al., 2020). It uses the two-dimensional WL isomorphic test and extends the order. The findings showed that, when compared to the conventional GNN method, the 2-WL GNN framework performs better at differentiating between the graphs.

A different study (Murphy, R. et al., 2019) assessed the relationship between GNN and the 1-dimension WL algorithm. According to the study, GNN and 1-WL have the same expression when it comes to distinguishing non-isomorphic subgraphs. This indicates that comparable types of results are produced by both methods. The k dimensional GNN, a high order GNN, was suggested in the study. In order to explore the characterisation of GNN in social networking and molecular graphs, high order GNN is essential. The theoretical investigation's conclusion shows that graph classification functioning may be used by higher order networks (Morris, C. et al., 2019).

III. METHODOLOGY

Because it mostly depends on secondary research, the research being conducted is non-experimental in character and does not require any form of experimentation. The non-experimental study's main goal is to acquire relevant data from large databases in order to enhance comprehension of GNNs, WL, and other current developments in the area. Non-experimental research is typically classified as descriptive or theoretical in nature. It employs appropriate techniques to characterize a phenomenon and explore the correlation among multiple variables that are the main subject of the study.

A. Research Methodology

The investigation of search techniques in graph neural networks and WL is done in this work using a quantitative research approach. The procedure is the first examination by which the research topic is understood and examined when an analysis is planned around a theoretical framework. Though quantitative investigations typically include theoretical frameworks. However, investigations that are quantitative in nature typically make use of theoretical frameworks. This method concentrates on investigating the subject from many theoretical perspectives. In order to successfully conclude the research's findings, the theoretical technique compares, synthesizes, analyzes, and draws conclusions from prior studies using inductive and deductive reasoning.

The study employed non-experimental methodology underneath a quantitative methodology to successfully fulfill the study goal and research objective. Generally speaking, non-experimental research is descriptive or theoretical in nature, using proper methods to describe a phenomenon and investigate the link between two or more variables that are the focus of the study. Non-experimental research did not change, manipulate, or have the capacity to change depending on the circumstances, in contrast to experimental research. The longitudinal research approach is suitably used in this study. The study's goal, to investigate graph neural networks are more potent than higher order WL algorithms and to analyze the expressive potential of graph neural networks, may be successfully and satisfactorily fulfilled by longitudinal research, which indicates that this approach is the most concentric and appropriate one from a research standpoint.

B. Data Analysis

The study focused on the theoretical rationale and inquiry into the graph neural network and WL algorithm. Analysis of graph neural networks' expressive capability and investigation of their potential superiority over higher order WL algorithms are the two prominent key goals of the study. The examination of the data can be categorized into multiple sub-parts in order to meet the goal and determine the research question. In every sub-part, various articles and academic endeavors are extensively reviewed to improve comprehension, pinpoint the analysis's findings, and determine whether the overall focus that emerges from existing research aligns with the goal and matter of the theme of this manuscript.

C. WL Algorithm and its Higher Order

The study (Morris, C et al., 2019) is exploring the WL algorithm and its degree. The 1- WL algorithm that belongs to a labeled graph (P, *l*), where P is a graph endowed with a color tuple. In every iteration $g \geq 0$, the 1-WL processed a node having color on it

e_l^g: D (P) → Σ, which relies on obtaining the coloring data from previous nodes. For iteration 0, let assume $e_l^0 = l$. For the iteration k > 0, the

$$e_l^{(g)}(r) \ = \ \text{HASH} \ \left\{ \left(e_l^{(g-1)}(r), \ \left\{ e_l^{(g-1)} \ (s) \mid \boldsymbol{s} \in N\ (r) \right\} \right) \right\} \qquad \ldots\ldots\ldots. (1)$$

Since the hash function was not employed in the previous iteration, it is a bijective map beyond coupling to generate a unique value in Sigma. The aforementioned procedure is used in parallel mode to both graphs P and H to determine if they are isomorphisms or not. When the number of duplicates or colored-in nodes in two graphs differs. The 1-WL method yielded the conclusion that neither of the graphs is isomorphic. Moreover, the test is over if the overall number of colors does not change between two rounds, indicating that the fundamentals of the image, $e_l^{(g-1)}$ and $e_l^{(g)}$are identical.

Assuming that the 1-WL method is either insufficient or incapable of differentiating between all non-isomorphic networks is a handy course of action. Nevertheless, according to Cai, J. Y. et al. (1992) and Babai, L., & Kucera (1979), it is powerful enough to detect isomorphisms for a wide range of graphs.

The research was subsequently extended by the study (Morris, C. et al., 2019) to investigate the capabilities of the k-dimensional WL algorithm and to compare it with the one-dimensional WL method. An extension of the 1-WL test, where the coloring tuples conform to D $(P)^{\text{k}}$ regardless of the nodes, is the k-dimensional WL method for the iterative corresponding to k.

That shows that the processing of the algorithm for coloring tuple $e_{l,k}^{(g)}$: D $(P)^{\text{k}} \rightarrow \Sigma$ .To identify the algorithm, let assume i-th neighbor.

$$N_i(\text{t}) \ = \ \left\{ \left(t_1, \ldots., t_{i-1}, r, t_{i+1}, \ldots, t_j \right) \mid r \in D\ (P) \right\} \ldots\ldots\ldots\ldots(2)$$

Of a i-tuple t = $(t_1, \ldots., t_k)$ in D $(P)^{\text{k.}}$ That is i-th neighbor N_i(t) of t is retrieved by altering the i- th element of t by each node from D (P). In iteration 0, the test labeling every k-tuple with its atomic kind i.e. two k-nodes t and t' in D $(P)^{\text{k}}$ producing the similar color if the mapping feature produces $q_b \rightarrow q'_b$ by induces a labeled node that describes isomorphism among two subgraphs as induced in the form of nodes *r* and *r*' respectively.

For recursively when the iteration k>0

$$E_i^{(g)}(q) \ = \ \text{HASH} \ \left\{ \left(e_{l,k}^{(g-1)}(q') \mid q' \in N_i(q) \right) \right\} \qquad \ldots. (3)$$

As well as

$$e_{l,k}^{(g)}(q) \ = \ \text{HASH} \ \left\{ \left(e_{k,l}^{(g-1)}(q), \left(E_1^{(g)}(q), \ldots., E_k^{(g)}(q) \right) \right) \right\} \ \ldots\ldots\ldots(4)$$

This shows two nodes q and q' along with exhibit distinct colors for the iteration g if there exist i in [1: i] in a manner that the number of neighbors of i-th term are q and q' , the nodes get colors with a distinct color for different nodes. Next, WL was introduced as a new dimension in the exam. Enhancing the k dimension increases the test's power and robustness in recognizing and differentiating non-isomorphic graphs at every iteration. In contrast to the k-WL technique, a k-value of 2 indicates well-identified and non-isomorphic graphs, as shown by (k + 1) WL (Cai, J.Y et al., 1992).

D. Graph Neural Network

A Graph E belongs to (P, Q) where r number of nodes comes under P as nodes set and edge set Q, directed and undirected. Every set of graph has an adjacency matrix A ∈ $\{0, 1\}^{\text{nxm}}$, and potential mode features X= $(\text{x}_1, \text{x}_2 \ldots\ldots\ldots \text{x}_\text{n})^{\text{T}} \in \text{R}^{\text{nxc}}$. Merging the adjacency matrix and node features in such a manner that they can be represented as E= (A, X). In condition when unavailability of explicit nodes can be identified that unfeatured graph E shall use as X= 11^{T}, here 1 is represented as n × 1 as a vector of ones. For a set of nodes as p ∈ P, then the neighboring nodes are denoted as P(p). Along With this, P |E| is represented as a node set of the given graphs.

GNN uses a neighborhood aggregation technique to encrypt every node in a manner such as p ∈ P as denoted vector h_p. This denotation of nodes is upgradable based on their iteration and collected neighbor information prior to implementing a nonlinear transformation approach on distinct neural layers.

$$\text{Message Passing: } h_r^{(i)} = MSG^{(i)} \left\{ \left(h_r^{(i-1)} : r \in N\ (p) \right) \right\}$$

$$\text{Aggregation: } a_p^{(i)} \ = \ AGG^{(i)} \left\{ \left(h_r^{(i-1)} : r \in N\ (p) \right) \right\}$$

$$\text{Upgrade: } h_p^{(i)} = UPD^{(i)} \left\{ \left(h_p^{(i-1)} : a_p^{(i)} \right) \right\}$$

Here, $h_p^{(0)} = x_v$, $h_r^{(i)}$, indicates the neighboring message, where $h_r^{(i)}$denotes the aggregation operation from the neighboring layers i, $MSG^{(i)}$, $AGG^{(i)}$ and $UPD^{(i)}$ represent the message passing, aggregation and upgrade operations of layer i. MPNNs are able to discover connections among neighboring nodes owing to their neighbor aggregating approach.

Similar to the one-dimensional WL method, the message passing architecture of graph neural networks modifies each node on each iteration based on the color of its neighboring node and the tuple. Through a recursive process of extending the neighbor of the root tuple, this computational process encodes an established subtree encompassed by all issues up to the final color. The effectiveness of graph neural networks in graph classification can be attributed to a phenomenon where a few graphs share information or have similar rooted subtrees that are more thoroughly described into similar classes. These classes are actually aligned with an inductive bias function that determines whether two graphs are similar if they have a comparable quantity of edges as well as vertices (Loukas, 2020).

Problems demanding relational thinking are a good fit for this inductive bias. The inability of MPNNs to discriminate among node identities and position, over-smoothing, over-compression, and restricted expressiveness capability are some of its drawbacks, despite their general effectiveness. Their incapacity to calculate many essential graph features (such as diameters, local clumping parameters, and shortest/longest loops and their incapacity to acquire knowledge of the graph architecture demonstrate their deficiency of expression capability. More capable GNNs are needed to cope with these issues wherever analyzing graph topology has become crucial, involving detecting a molecule's chemical characteristics and resolving stochastic optimization difficulties.

E. Higher order GNN model to resolve the above mention challenges-

Two different neighborhoods that have several pair feature vectors at the same location never contain the maximal power of a GNN. According to this, the GUNN's overall operation has to be injective. This demonstrates how a neural network may represent a group of aims spanning several sets and determine whether or not they can perform injective multi-set operation.

Suppose (P, l) be the labeled graph, in every k-dimensional graph neural network strata, let the iteration g≥ 0, process of feature vector $f_k^{(g)}\,(u)$ for every set u belongs to k-set in the $[\mathrm{D}(P)]^k$. Initially, when the iteration g = 0 it was indicated that by replacing $f_k^{(g)}\,(u)$ to $f^{(iso)}\,(u)$, which is a one- hot encryption, by the labeling l of an isomorphism kind of graph as P[u]. In every stratum of GNN, noble feature is extracted as

$$f_k^{(g)}\,(u) = \Theta\Big\{\big(\,f_k^{(g-1)}\,(u).V_1^{(g)} + \textstyle\sum_{r\,\in\,N_H(u)\,\cup\,N_G(u)} \quad f^{(g-1)}\,(r).V_2^{(g)}\;\big)\Big\}$$

To further improve understanding and ascertain the importance of local and global neighbors, the summation might also be divided into two smaller sections using varying parameter matrices. Research by Morris, C. et al. (2019) suggested using locally or regionally distributed graph neural networks, that can filter out universal neighbors, to organize multilayer graph neural networks onto large databases and avoid excessive fitting. The initial investigation focuses on the unique characteristics of maximal capacity for the standard class of GNN framework. When mapped to different representations in the embedding location, a GNN with the greatest robustness level exhibits unique graph structure properties. One way to get around the difficulties of the graph isomorphic behavior problem is to link graphs with various embeddings. The paper states that although non-isomorphic networks may be expressed separately on a map of a region, isomorphic graphs might be presented identically.

$$f_{k,H}^{(g)}\,(u) = \Theta\Big\{\big(\,f_{k,H}^{(g-1)}\,(u).V_1^{(g)} + \textstyle\sum_{r\,\in\,N_G(u)} \quad f_{k,H}^{(g-1)}\,(r).V_2^{(g)}\;\big)\Big\}$$

if k ≥0, let the iteration g≥ 0, process of feature vectors having weight $V^{(g)}$, the GNN architecture that may possess $[\mathrm{D}(P)]^k$.

$$e_{u,k,l}^{(g)} \equiv f_k^{(g)}$$

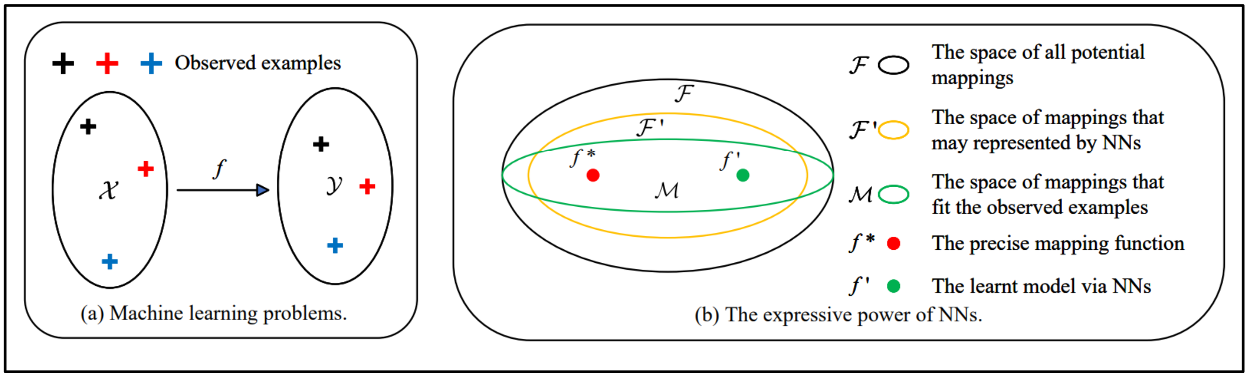
The aforementioned explanations and theorems have led to the conclusion that the one-dimensional WL algorithm outperforms a wide range of graph neural network frameworks in terms of strength. While it presents a potentially useful technique, it also demonstrates that GNNs can effectively detect known isomorphic subgraphs, just like well-calculated WL algorithms do. Another benefit of k-dimensional graph neural networks, a generalized version of graph neural networks depending on k-WL databases.

In comparison to 1-GNN, this suggested technique offers a more potent, reliable, and practical learning strategy that operates consistently well for cutting-edge neural architectures based on big databases (Morris, C et al., 2017; Hamilton, W et al., 2017). Such a training technique is realistic and promising. Thus, it can be inferred from the preceding equation that the characteristics are learned recursively in a throughout its entirety pattern from one dimension to k. In addition to showing that the suggested hierarchical variation design is capable, it also shows that it has greater compatibility with the k-dimensional WL algorithm and discloses the counterparts of k-dimensional graph neural networks. Despite this, the investigation was unable to apply the model to actual network association and associated structure since the technique depends on conceptual implementation.

The study is anticipating an enhanced form of it, known as k-GNN, or higher order GNN. The research determines the driving force behind k-GNN, which is linked to k-WL's advancement. In order to explore the characterisation of GNN in social networking and molecular graphs, high order GNN is essential. The theoretical investigation's conclusion shows that graph classification functions may be used by higher order networks. Through the use of machine learning, it attempts to convey a perceptive perspective and its efficacy.

F. Need of Expressiveness

It is possible to abstract all neural network issues as training a mapping f ∗ between the space of features X to the target space Y. Typically, f ∗ is estimated by maximizing certain parameters θ in a model f(θ). Since f∗ is usually unknown beforehand in practice, it is desirable for f(θ) to resemble a large range of f∗ as closely as possible. This range's estimated width is known as the model's expressive power, and it provides a crucial gauge of the model's ability to communicate as seen in Fig 1 (a).


Observed examples
$\mathcal{X}$
f
$\mathcal{Y}$
$\mathcal{F}$
$\mathcal{F}'$
f^*
f'
$\mathcal{M}$
$\mathcal{F}$ The space of all potential mappings
$\mathcal{F}'$ The space of mappings that may represented by NNs
$\mathcal{M}$ The space of mappings that fit the observed examples
f^* ● The precise mapping function
f' ● The learnt model via NNs
(a) Machine learning problems.

(b) The expressive power of NNs.

Figure 1: Expressiveness of NN (Wang, Q et al., 2023).

The remarkable expressive capacity of neural networks (NNs) is demonstrated by their capacity to estimate all perpetual functions. More precisely, this capacity is the characteristic embedding ability, as illustrated in Fig. 1(b), meaning the capacity to incorporate data from the field of features X through the target domain Y produced by any function that is continuous. Some succeed, in comparison, inquire about the expressiveness of GNNs, and these have proven that they perform noticeably better in a range of tasks related to applications because of their exceptional characteristic embedding ability. This can be attributed to GNNs exhibiting strong expressive strength due to their outstanding feature embedding competence.

E. Strengthening GNN Expressiveness

The research investigation (Xu, K., et al., 2018) analyzed different nodes situated in the same place in an embedding space to ascertain the expressive capability of graph neural networks. The formulation of the most powerful GNN with two nodes in the same place is only possible if the nodes are part of the same tree structures and share attributes with their associated nodes. Due to the fact that the sub-tree structure may identify or detect recursive properties of nearby nodes. In order to determine the maximal power of gnn, the present study (Xu, K., et al., 2018) aims to demonstrate how two subsets that belong to the identical and integrated position may be related with each other. This suggests that the GNN's overall operation has to be injective. This demonstrates how a neural network's aggregation function, expressed as a class of objectives spanning several sets, may determine whether or not a neural network has the ability of injective multi-set functioning.

The findings by Morris et al. (2019) and Xu et al. (2018) that show the remarkable parallels between neighbor label aggregation in the WL test and neighborhood aggregation in GNNs lead to this conclusion. Since using injection functions makes GNNs equipped with the WL test in terms of expressive power, a decision of aggregation parameters for GNNs is important. GNNs exceed the 1-WL test in terms of expressive capacity when feature embedding is taken into account. The capacity of GNNs to identify any graph in which at least one node demonstrates a distinctive trait is compellingly demonstrated by Kanatsoulis et al. (2022).

IV. RESULTS

According to the study, GNN and 1-WL have the same expression when it comes to distinguishing non-isomorphic subgraphs. The k-dimensional GNN, a high order GNN, was suggested in the study. In order to explore the characterisation of GNN in social networking and molecular graphs, high order GNN is essential. The result of the theoretical study shows that graph classification functions may be used by higher order networks. It makes an effort to demonstrate a perceptive perspective and its efficacy by applying a machine learning methodology. The data examines the expressiveness and capabilities of the WL algorithm as well as the efficacy of supervised learning techniques in detecting node categorization. It demonstrates that, when computed effectively, the WL algorithm and GNN have comparable abilities and powers to discriminate known isomorphic subgraphs. An additional benefit of k-dimensional graph neural networks, which are a modified type of graph neural networks that depend on the k-WL database. By providing a straightforward learning strategy, this suggested methodology consistently performs successfully and shows promise for state-of-the-art neural architecture based on huge databases. It behaves more robustly and powerfully than 1-GNN.

According to the research (Morris, C. et al., 2019), regional or local graph neural networks may be used to filter out global neighbors. First, the normal class of GNN framework's distinctive attributes of maximal capability are investigated. The unique graph structure properties of a maximum resilient GNN are mapped to differentiate representation in the embedded location. The graph isomorphism problem may be solved by using this ability to connect different graphs with unique embeddings. According to the study, non-isomorphic graphs may be shown with unique properties, whereas isomorphic graphs can be mapped with a comparable representation.

Using a particular variation of k-WL, Morris et al. (2019) introduced k-GNN, which gains features over segments of the graphs on k vertices and is utterly inferior to k-WL. Even though Maron et al. (2018, 2019) developed k-order GNNs that are equally expressive as k-WL, they also produced simplified 2-order GNNs that are just as efficient as 3-WL. Higher level GNNs were proposed by Morris et al. (2020; 2022) and Zhao et al. (2022) by considering just a portion of all k-tuples, particularly those that are local or related to forming subsets of graphs with certain linked nodes. Even while these GNN architectures are all clearly stronger than 1-WL, they also have the same k-drawbacks as 1-WL, including an odd structure and high computational costs, which means they are not appropriate for real-world jobs, particularly ones where graphs seem to be very large. Unlike previous research, the suggested GNN system is not built using traditional k-WL. Essentially, the N-WL algorithm, is a node iterative technique that leverages high-order encouraged subsections in its d-hop neighborhood; that is, it integrates a subgraph with color tuples into the node coloring instead of employing k-tuple coloring with the same aspect to spread a k-tuple color palette. The N-WL method considers adjacent structures that are d-hops larger than its own, using the exact same locality as conventional GNNs.

V. CONCLUSION

The study achieves this goal by examining the theoretical framework of graph neural networks and determining various attributes, with a particular emphasis on the expressiveness of GNNs. In order to handle circumstances such as graph isomorphism, the study places a strong emphasis on investigating GNN and its relationship with the WL algorithm. During examination originally the research started with discovering how GNN is strong when paired with WL test, and progressed onto greater-order GNN and k-dimensional WL test. In addition, the research also highlights the difficulties and constraints that come with using traditional GNN and WL techniques, which may be overcome by using more sophisticated GNN and WL methods.

REFERENCES

[1] Bevilacqua, B., Frasca, F., Lim, D., Srinivasan, B., Cai, C., Balamurugan, G., ... & Maron, H. (2021). Equivariant subgraph aggregation networks. arXiv preprint arXiv:2110.02910.

[2] Bianchi, F. M., & Lachi, V. (2023). The expressive power of pooling in Graph Neural Networks. arXiv preprint arXiv:2304.01575.

[3] Bouritsas, G., Frasca, F., Zafeiriou, S., & Bronstein, M. M. (2022). Improving graph neural network expressivity via subgraph isomorphism counting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(1), 657-668.

[4] Chen, Z., Xu, J., Alippi, C., Ding, S. X., Shardt, Y., Peng, T., & Yang, C. (2021). Graph neural network-based fault diagnosis: a review. arXiv preprint arXiv:2111.08185.

[5] Grohe, M. (2021, June). The logic of graph neural networks. In 2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS) (pp. 1-17). IEEE.

[6] Hang, M., Neville, J., & Ribeiro, B. (2021, July). A collective learning framework to boost gnn expressiveness for node classification. In the International Conference on Machine Learning (pp. 4040-4050). PMLR.

[7] Joshi, C. K., Bodnar, C., Mathis, S. V., Cohen, T., & Lio, P. (2023). On the expressive power of geometric graph neural networks. arXiv preprint arXiv:2301.09308.

[8] Kanatsoulis, C. I., & Ribeiro, A. (2022). Graph Neural Networks Are More Powerful Than We Think. arXiv preprint arXiv:2205.09801.

[9] Kumar, A., Das, S., Roy, S., Maity, B., & Dasgupta, A. (2023). Improving Expressivity of Graph Neural Networks using Localization. arXiv preprint arXiv:2305.19659.

[10] Li, P., & Leskovec, J. (2022). The Expressive Power of Graph Neural Networks. In Graph Neural Networks: Foundations, Frontiers, and Applications (pp. 63-98). Springer, Singapore.

[11] Liu, M., Yu, H., & Ji, S. (2022). Empowering GNNs via Edge-Aware Weisfeiler-Lehman Algorithm.

[12] Liu, M., Yu, H., & Ji, S. (2022). Your neighbors are communicating: Towards powerful and scalable graph neural networks. arXiv preprint arXiv:2206.02059.

[13] Loukas, A. (2020). How hard is it to distinguish graphs with graph neural networks? Advances in neural information processing systems, 33, 3465-3476.

[14] Maron, H., Ben-Hamu, H., & Lipman, Y. (2019). Open problems: Approximation power of invariant graph networks. In NeurIPS 2019 Graph Representation Learning Workshop, 2019a. URL https://grlearning. github. io/papers/31. pdf (Vol. 2, No. 8, p. 31).

[15] Maron, H., Ben-Hamu, H., Serviansky, H., & Lipman, Y. (2019). Provably powerful graph networks. Advances in neural information processing systems, 32.

[16] Michel, G., Nikolentzos, G., Lutzeyer, J. F., & Vazirgiannis, M. (2023, July). Path neural networks: Expressive and accurate graph neural networks. In the International Conference on Machine Learning (pp. 24737-24755). PMLR.

[17] Morris, C., Fey, M., & Kriege, N. M. (2021). The power of the weisfeiler-leman algorithm for machine learning with graphs. arXiv preprint arXiv:2105.05911.

[18] Morris, C., Rattan, G., & Mutzel, P. (2020). Weisfeiler and Leman go sparse: Towards scalable higher-order graph embeddings. Advances in Neural Information Processing Systems, 33, 21824-21840.

[19] Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., & Grohe, M. (2019, July). Weisfeiler and leman go neural: Higher-order graph neural networks. In Proceedings of the AAAI conference on artificial intelligence (Vol. 33, No. 01, pp. 4602-4609).

[20] Ni, J., Qin, J., & Huang, D. (2021, October). Identity-aware graph memory network for action detection. In Proceedings of the 29th ACM International Conference on Multimedia (pp. 3437-3445).

[21] Nunes, M., & Pappa, G. L. (2020). Neural architecture search in graph neural networks. In Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I 9 (pp. 302-317). Springer International Publishing.

[22] Schnake, T., Eberle, O., Lederer, J., Nakajima, S., Schütt, K. T., Müller, K. R., & Montavon, G. (2021). Higher-order explanations of graph neural networks via relevant walks. IEEE transactions on pattern analysis and machine intelligence, 44(11), 7581-7596.

[23] Vignac, C., Loukas, A., & Frossard, P. (2020). Building powerful and equivariant graph neural networks with structural message-passing. Advances in Neural Information Processing Systems, 33, 14143-14155.

[24] Wang, Q., Chen, D. Z., Wijesinghe, A., Li, S., & Farhan, M. (2023, February). $\mathscr {N} $-WL: A New Hierarchy of Expressivity for Graph Neural Networks. In The Eleventh International Conference on Learning Representations.

[25] Wang, Z., Cao, Q., Shen, H., Bingbing, X., Zhang, M., & Cheng, X. (2022, December). Towards Efficient and Expressive GNNs for Graph Classification via Subgraph-aware Weisfeiler-Lehman. In The First Learning on Graphs Conference.

[26] Wei, L., Zhao, H., He, Z., & Yao, Q. (2023). Neural Architecture Search for GNN-based Graph Classification. ACM Transactions on Information Systems.

[27] Xu, K., Hu, W., Leskovec, J., & Jegelka, S. (2018). How powerful are graph neural networks? arXiv preprint arXiv:1810.00826.

[28] You, J., Gomes-Selman, J. M., Ying, R., & Leskovec, J. (2021, May). Identity-aware graph neural networks. In Proceedings of the AAAI conference on artificial intelligence (Vol. 35, No. 12, pp. 10737-10745).

[29] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., ... & Sun, M. (2020). Graph neural networks: A review of methods and applications. AI Open, 1, 57-81.