The Polynomial Counting Capabilities of Message Passing Neural Networks

Marco Sälzer[1], **Pascal Bergsträßer**[1], **Anthony W. Lin**[1,2]

[1]RPTU University Kaiserslautern-Landau

[2]Max Planck Institute for Software Systems (MPI-SWS)

{marco.saelzer, pbergstr}@rptu.de, awlin@mpi-sws.org

Abstract

The counting power of Message Passing Neural Networks (MPNN) has been the subject of many recent papers, showing that they can express logic that involves counting up to a threshold or more generally satisfy a linear arithmetic constraint. In this paper, we study the counting capabilities of MPNN beyond linear arithmetic, primarily utilising local and global mean aggregations. In particular, our goal is to tease out conditions required to express extensions of graded modal logic with polynomial counting constraints. We show that global polynomial counting constraints in node-labelled graphs can be checked using mean MPNN under mild assumptions. Checking local constraints is also possible, if we consider formulas with no nested modalities and additionally either (i) permit sum/max aggregations, or (ii) only restrict to regular graphs. We also show how formulas with nested modalities can be captured by mean MPNN over graphs with tree-like structures and similar assumptions.

1 Introduction

Message Passing Neural Networks (MPNN) are a class of neural network-based models trained for solving tasks over structured data, i.e., graphs. By design, MPNN effectively utilises the structure of the input data, in stark contrast with other approaches such as using handcrafted encoding followed by classical feedforward neural networks (FNN). This has led them to become one of the go-to models for addressing tasks such as drug discovery (Bongini, Bianchini, and Scarselli 2021), a broad range of knowledge graph applications (Zhou et al. 2020), recommender systems (Wu et al. 2022), or natural science applications (Kipf et al. 2018; Shlomi, Battaglia, and Vlimant 2020).

Sparked by these and other promising applications of MPNN, recent years have seen a growing line of research thoroughly investigating the expressive capabilities of MPNN using formal frameworks such as logics. We provide a comprehensive overview in Section 1.1. Starting with comparisons to classical logics like fragments of first-order logic with counting (up to a threshold) done by (Barceló et al. 2020), later works by (Nunn et al. 2024; Benedikt et al. 2024) have explored tailored extensions of such classical logics by arithmetic expressions to tighten the characterisations and infer the complexity of verification tasks related to MPNN. However, all of these works

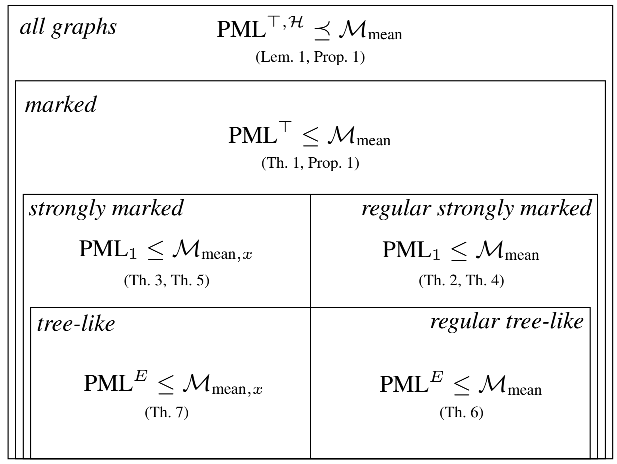


Figure 1: Overview of results. The MPNN class $\mathcal{M}_{\mathrm{mean},x}$ refers to MPNN that use mean and either sum or max as aggregation functions, $\leq$ refers to the notion of recognition with some certainty defined in Sec. 2, and $\preceq$ denotes the lowerbound on the certainty of MPNN in checking modal formulas which is exclusive to Lem. 1.

have one thing in common: the logics employed for comparison with MPNN are confined to extensions by *linear integer arithmetic*. For example, such logics could express relationship *a node has the same number of blue-coloured neighbours and green-coloured neighbours*.

In this paper, our goal is to study the capability of MPNN to express counting properties *beyond* linear integer arithmetic. *Counting* plays an important role especially owing to connection to *graph kernels* (e.g. see (Kriege, Johansson, and Morris 2020; Shervashidze et al. 2011; Jaakkola et al. 2025)) — such as Weisfeiler Leman kernel — for classifying graphs by counting the number of occurrences of "neighborhoods" in the graphs. In the graph classification problem, a more general polynomial counting constraint has featured (e.g. see (Zhang, Liu, and Jiang 2024)), which is done by applying a polynomial kernel and then Support Vector Machines (SVM). The ability of MPNN to express polynomial counting properties would show essentially that it generalizes the aforementioned technique for graph classification. *In this paper, we identify conditions where MPNN can express polynomial counting properties.*

Informally, an MPNN uses a predefined number of message-passing rounds where each node in a graph exchanges information along two streams: with its direct neighbours and with all nodes in the graph. Incoming information is then aggregated per node using different functions, such as sum, mean, or max, and is combined with the current state of a node using a learnable function, usually represented by an FNN, which results in an updated state for each node. Prior studies on characterising the expressive capabilities of MPNN using logic have primarily focused on MPNN that operate over node-labelled graphs and utilise sum to aggregate incoming information. We identify the class of MPNN using mean aggregation as a candidate to posess polynomial counting capabilities[1]. In particular, we demonstrate that under certain assumptions mean MPNN are able to recognise different fragments of Peano modal logic (PML), a logic that allows for different forms of modalities, local as well as global ones, and polynomial counting constraints (not only linear ones). In particular, PML strictly extends Graded Modal Logic (Barceló et al. 2020). An overview of our resuls is provided in Figure 1. In general, we focus on identifying minimal assumptions about the class of inputs to enable mean MPNN to recognise certain fragments of PML. Interestingly, we also identify settings where allowing additional sum or max aggregations eases the otherwise required assumptions.

Conditions. Our conditions enabling GNN to perform polynomial counting (see Figure 1) amount to restricting the input graphs. These include (i) adding a special marked node, (ii) regularity, and (iii) restricting input graphs to tree-like structures. We briefly discuss their practical implications below.

The "marked node" assumption needs no extensive justification in practice. Specifically, when using an MPNN for a node classification task, a simple preprocessing step marking the node under consideration ensures this property holds without substantially altering the datapoint's semantics. Similarly, the same can be ensured in the training phrase. The benefits of adding such preprocessing steps are theoretically investigated in multiple studies (e.g. (Zhang et al. 2021; Abboud et al. 2021)); in fact, variants thereof are used in GNN models such as SEAL or ID-GNN. Incidentally, a similar assumption is also used for textual data (e.g. for transformers and RNN), in that an end-of-string symbol and/or a beginning-of-string symbol are often present and beneficial (e.g. see (Sälzer et al. 2026; Yang, Chiang, and Angluin 2024)).

Conditions (ii) and (iii) requiring regularity or tree-like structures can be more restrictive. However, the assumption of, for example, tree-like inputs is justified by applications like learning tasks on semi-structured data such as those in XML, HTML and JSON, which are abundant in practice. In particular, one can think of classification problems of such semi-structured data. In general, counting in trees are certainly relevant and appear in various works (e.g. (Seidl, Schwentick, and Muscholl 2008; Hague, Lin, and Hong 2019)). For example, as elaborated in detail in (Hague, Lin, and Hong 2019), the node selector language in Cascading Style Sheets (CSS) — the de facto language for styling web documents — supports counting the number of children of a matched node.

Organization. The paper is structured as follows. In Section 2, we define the necessary fundamentals, such as MPNN, PML, and what it means for the former to recognise the latter. In Section 3, we investigate the extent to which MPNN can recognise shallow fragments of PML, meaning that we do not nest any modal subformulas. Subsequently, we dedicate Section 4 to the analysis of the capabilities of MPNN in relation to PML fragments with nesting. Finally, in Section 5, we summarise and discuss the next steps. Some formal arguments are of a rather technical nature and are used repeatedly; thus, we have omitted them in the main part and deferred them to Appendix A.

1.1 Related Work

The seminal work by (Barceló et al. 2020) investigates the expressive capabilities of MPNN using sum aggregation exclusively via fragments of the logic FOC_2, which denotes the two-variable fragment of first-order logic with counting. Two prominent results are that all graded modal logic classifiers are captured by local MPNN with sum aggregation and that full FOC_2 is captured by MPNN also using global sum aggregation. These results are succinctly summarised and placed in context of characterisations based on the Weisfeiler-Leman algorithm in (Grohe 2021), though the latter is only loosely connected to this paper. Subsequent studies such as done by (Benedikt et al. 2024) and (Nunn et al. 2024) look at extensions of classical logics by means of linear arithmetic to further characterise the expressive capabilities of MPNN with sum aggregation. Notably, besides results about the complexity of reasoning about such MPNN, these studies deliver upper bounds on their expressiveness, but only under certain assumptions such as eventually constant activation functions.

Going beyond plain sum MPNN, (Cucala et al. 2023) show that monotonic MPNN using either maximum or sum aggregation can be captured by Datalog programs. Similarly, (Ahvonen et al. 2024) examine the expressive capabilities of recursive MPNN, those with input-dependent depth, in relation to properties expressible in monadic second-order logic (MSO). Notably, this study considers differences arising from assumptions about the underlying arithmetic, such as real arithmetic versus floating-point arithmetic, showing that assuming the latter leads to generally weaker MPNN. (Rosenbluth, Tönshoff, and Grohe 2023) compared differences in the expressive capabilities of MPNN exclusively using sum, mean or max aggregation. A work closely related to the setting of this paper is the recent study by (Schönherr and Lutz 2025), which examines the logical expressiveness of MPNN with mean aggregation. They derive sev-

[1](Schönherr and Lutz 2025) recently investigated the expressive capabilities of mean MPNN in relation to specific modal logics and monadic second-order logic (MSO), though they do not consider genuine polynomial properties.

eral results based on modal logics, such as ratio modal logic (RML), which allows for expressing proportional properties of neighbourhoods.

We remark that in research on capturing the expressive capabilities of transformer architectures, there is also a line looking into their logical expressiveness (Chiang, Cholak, and Pillay 2023; Barceló et al. 2024). While this appears only loosely related at first glance, MPNN with powerful aggregations, such as graph attention networks (GAT) (Veličković et al. 2018), can be considered as a generalisation, leading to many similarities in these types of results. Notably, the very recent study (Sälzer et al. 2026) examines the polynomial counting capabilities of transformers using average hard attention, employing constructions similar to ours.

2 Preliminaries

We denote vectors using bold symbols such as $\boldsymbol{x}$, $\boldsymbol{y}$, or $\boldsymbol{z}$. We use $\boldsymbol{x}_\perp$ to denote the element in the last dimension of vector $\boldsymbol{x}$. We denote *multisets*, which are sets containing duplicate elements, using the notation $\{\{\cdot\}\}$.

A *(directed) graph* G is a tuple (V, E, L) where V is a finite set of vertices, $E \subseteq V \times V$, and $L \colon V \to \{0,1\}^k$ is a labelling function. We refer to the different dimensions of L as *colours* and say that G *has k colours*. We denote the class of all directed graphs by $\mathcal{G}$. Let $v \in V$. We define the *ingoing (outgoing) neighbourhood of* v by $\text{neigh}_{\text{in}}(v) = \{u \mid (u, v) \in E\}$ ($\text{neigh}_{\text{out}}(v) = \{u \mid (v, u) \in E\}$), and the general *neighbourhood of* v by $\text{neigh}(v) = \text{neigh}_{\text{in}}(v) \cup \text{neigh}_{\text{out}}(v)$. We call G *ingoing (outgoing) regular* if for all $u, v \in V$ we have $|\text{neigh}_{\text{in}}(v)| = |\text{neigh}_{\text{in}}(u)|$ ($|\text{neigh}_{\text{out}}(v)| = |\text{neigh}_{\text{out}}(u)|$). If both are given, we refer to G simply as *regular*. We call a pair (G, v) a *pointed graph* and call v the *focus*. We call a node v in G *marked* if there is dimension i of the labelling L such that for all $u \in V$ we have $L(u)_i = 1$ if and only if $u = v$. Let $\mathcal{G}'$ be any class of directed graphs. We denote by $(\mathcal{G}')^p$ for $p \in \mathbb{N}$ the class of all graphs $G \in \mathcal{G}'$ that contain a node v that is marked by colour p. We call v *strongly marked* if its marked and $v \in \text{neigh}(v)$. Likewise, we denote by $(\mathcal{G}')^p_\bullet$ for $p \in \mathbb{N}$ the class of all pointed graphs (G, v) where $G \in \mathcal{G}'$ and focus v is marked by colour p. Let $v, v' \in V$. We call a sequence $u_0, \ldots, u_{n-1}$ with $u_0 = v$, $u_{n-1} = v'$, and $(u_i, u_{i+1}) \in E$ or $(u_{i+1}, u_i) \in E$ for all $0 \leq i < n-1$ a *walk of length* $n-1$ *from* v *to* v'. If such a walk exists from v to v', we say that v' is at a walking distance $(n-1)$ from v.

Feedforward Neural Networks with ReLU

A *Feedforward Neural Network (FNN) with ReLU activations* is built from basic computational units, employing the ReLU activation $\mathit{relu}(x) = \max(0, x)$, called a *ReLU neuron*. A ReLU neuron is defined as a mapping $\mathbb{Q}^m \to \mathbb{Q}$ by $v(x_1, \ldots, x_m) = \mathit{relu}(b + \sum_{i=1}^m w_i x_i)$, where $w_i \in \mathbb{Q}$ are the *weights* and $b \in \mathbb{Q}$ is the *bias*. An *FNN layer* ℓ consists of a tuple of ReLU neurons $(v_1, \ldots, v_n)$, each having the same input dimension, realizing a mapping $\mathbb{Q}^m \to \mathbb{Q}^n$. A complete *FNN* N is then structured as a sequence of such layers $(\ell_1, \ldots, \ell_k)$ where the output dimension of each layer ℓ_i matches the input dimension of the next layer ℓ_{i+1}. The network N calculates a function from $\mathbb{Q}^{m_1}$ to $\mathbb{Q}^{n_k}$, given by $\mathcal{N}(x_1, \ldots, x_{m_1}) = \ell_k(\ldots \ell_1(x_1, \ldots, x_{m_1}) \ldots)$.

We primarily use FNN as components within message passing neural networks, as defined below. To do so, we frequently employ the term *gadget* to describe combinations of ReLU nodes that perform a specific function. When these gadgets are integrated into larger FNN, it is always given that this integration is feasible in the sense of forming a well-defined FNN.

Message Passing Neural Networks

A *message passing neural network* (MPNN) M (Barceló et al. 2020; Gilmer et al. 2017) is a tuple $(l_1, \ldots, l_L)$ where each l_i is a so-called *layer* given by $l_i = (\mathit{comb}_i, (\mathit{loc}_{\text{in}})_i, (\mathit{loc}_{\text{out}})_i, \mathit{glob}_i)$, consisting of a *combination function* comb_i, two *local aggregation* $(\mathit{loc}_x)_i$, one for ingoing and one for outgoing edges, and a *global aggregation* glob_i. Let $G = (V, E, L)$ be a graph. MPNN M computes a new *state* $\boldsymbol{x}_v^{(i)}$ for each node $v \in V$ by $\boldsymbol{x}_v^{(0)} = L(v)$, and $\boldsymbol{x}_v^{(i)} = \mathit{comb}_i(\boldsymbol{x}_v^{(i-1)}, \{(\mathit{loc}_x)_i(\{\{\boldsymbol{x}_u^{(i-1)} \mid u \in \text{neigh}_x(v)\}\})\}_{x \in \{\text{in,out}\}}, \mathit{glob}_i(\{\{\boldsymbol{x}_u^{(i-1)} \mid u \in V\}\}))$. We also write $M(G, v)$ for the final state $\boldsymbol{x}_v^{(L)}$ of node $v \in V$. We generally assume that combination functions comb_i are realised by feedforward neural networks (FNN) using ReLU activations. Let $\mathcal{M}$ be a class of arbitrary MPNN. In particular, we consider the class of all MPNN M where all local and global aggregations are realised by

$$f(\{\{\boldsymbol{x}_1, \ldots \boldsymbol{x}_n\}\}) = \frac{1}{|\{\{\boldsymbol{x}_1, \ldots \boldsymbol{x}_n\}\}|} \sum_{i=1}^{n} \boldsymbol{x}_i,$$

We also consider MPNN that use sum or max aggregations, defined by

$$f(\{\{\boldsymbol{x}_1, \ldots \boldsymbol{x}_n\}\}) = \sum_{i=1}^{n} \boldsymbol{x}_i,$$
$$f(\{\{\boldsymbol{x}_1, \ldots \boldsymbol{x}_n\}\}) = \max(\boldsymbol{x}_1, \ldots \boldsymbol{x}_n),$$

where max means taking the dimensionwise maximum. Let $A \subseteq \{\text{mean}, \text{sum}, \text{max}\}$. In the case that $n = 0$, we always assume that $f(\{\{\,\}\}) = \mathbf{0}$ of matching dimensionality. We refer to the class of MPNN that use aggregations from A by $\mathcal{M}_A$.

Peano modal logic

We consider modal logics extended with Peano arithmetic. We begin by defining Peano arithmetic constraints over graphs. Let X be a countable set of variables. A *Peano formula* ψ is defined by the grammar

$$\psi ::= \zeta \leq c \mid \psi \wedge \psi \mid \neg \psi$$
$$\zeta ::= x \mid c \cdot \zeta \mid \zeta + \zeta \mid \zeta \cdot \zeta$$

where $x \in X$ and $c \in \mathbb{Z}$. We call ζ a *(Peano) term*. We use $\psi(x_1, \ldots, x_m)$ or $\zeta(x_1, \ldots, x_m)$ to denote that x_1 to x_m are the variables occurring in ψ or ζ, respectively. The *semantics of* $\psi(x_1, \ldots, x_m)$, denoted by $[\![\psi(x_1, \ldots, x_m)]\!]$, are all tuples $(n_1, \dots, n_m) \in \mathbb{N}^m$ that satisfy ψ, written $n_1, \dots, n_m \models \psi$, in the obvious sense. We generally assume that Peano terms are given in the normalised form $\sum_{i=1}^{K} a_i \cdot \prod_{j=1}^{k_i} x_{i_j} \le b$, refer to $a_i \cdot \prod_{j=1}^{k_i} x_{i_j}$ as a *monomial* and to k_i as the *degree* of this monomial. Then, we call k *the degree of* ψ, denoted by $deg(\psi)$, if it is the maximum of the degrees of all monomials occurring in ψ.

Let P be a countable set of propositions. A *Peano modal logic (PML)* formula φ is defined by the grammar

$$\pi ::= id \mid E_{\text{in}} \mid E_{\text{out}} \mid \top$$

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \langle \pi_1, \dots, \pi_m \rangle_{\psi(x_1,\dots,x_m)}(\varphi_1, \dots, \varphi_m)$$

where $p \in P$ and $\psi(x_1, \dots, x_m)$ is a Peano formula as defined above. We call π a *modality*, and refer to $\langle \pi_1, \dots, \pi_m \rangle_{\psi(x_1,\dots,x_m)}(\varphi_1, \dots, \varphi_m)$ as *modal formulas*.

We define the *modal depth* of a PML formula inductively by $md(p) = 0$, $md(\neg\varphi) = md(\varphi)$, $md(\varphi_1 \wedge \varphi_2) = \max(md(\varphi_1), md(\varphi_2))$, and $md(\langle \pi_1, \dots, \pi_m \rangle_\psi(\varphi_1, \dots, \varphi_m)) = 1 + \max(md(\varphi_1), \dots, md(\varphi_m))$. Likewise, we define the *set of subformulas of* φ, denoted by $sub(\varphi)$, inductively by $sub(p) = \{p\}$, $sub(\neg\varphi) = sub(\varphi) \cup \{\neg\varphi\}$, $sub(\varphi_1 \wedge \varphi_2) = sub(\varphi_1) \cup sub(\varphi_2) \cup \{\varphi_1 \wedge \varphi_2\}$, and $sub(\langle \pi_1, \dots, \pi_m \rangle_\psi(\varphi_1, \dots, \varphi_m)) = \{\langle \pi_1, \dots, \pi_m \rangle_\psi(\varphi_1, \dots, \varphi_m)\} \cup \bigcup_{i=1}^{m} sub(\varphi_i)$. We denote by Mod_φ^k the set of all modalities occurring in modal subformulas of φ with modal depth k. We extend the definition of degree from Peano formulas to PML formulas by defining the *degree of* φ, denoted by $deg(\varphi)$ as the maximum of all degrees of Peano formulas ψ in φ. We define $Trace_\varphi^k \subseteq \{E_{\text{in}}, E_{\text{out}}\}^k$ for all $1 \le k \le md(\varphi)$ as the set of all sequences $E_0, \dots, E_{k-1}$ for which there exists a sequence of subformulas $\varphi_0, \dots, \varphi_k$ of φ such that for all $i < k$ there is $1 \le j \le m_i$ such that $\varphi_i = \langle \pi_1, \dots, \pi_{m_i} \rangle_\psi(\psi_1, \dots, \psi_{m_i})$ with $\varphi_{i+1} \in sub(\psi_j)$ and $E_i = \pi_j$ and $md(\varphi_i) = md(\varphi) - i$.

The *semantics of a PML formula* φ, denoted by $[\![\varphi]\!]$, is the set of all pointed graphs (G, v) with $G = (V, E, L)$ that satisfy φ, written $G, v \models \varphi$, which is defined as follows:

$$\begin{aligned} G, v \models p_i &\quad \text{iff} \quad L(v)_i = 1, \\ G, v \models \neg\varphi &\quad \text{iff} \quad G, v \not\models \varphi, \\ G, v \models \varphi_1 \wedge \varphi_2 &\quad \text{iff} \quad G, v \models \varphi_1 \text{ and } G, v \models \varphi_2, \text{ and} \end{aligned}$$

$G, v \models \langle \pi_1, \dots, \pi_m \rangle_{\psi(x_1,\dots,x_m)}(\varphi_1, \dots, \varphi_m)$ iff

- for all $i \in \{1, \dots, m\}$ there are exactly $n_i \in \mathbb{N}$ nodes $u \in V$ such that $u \in [\![\pi_i]\!]_v^G$ and $G, u \models \varphi_i$, and
- $n_1, \dots, n_m \models \psi(x_1, \dots, x_m)$,

where the *semantics of modality* π *given* (G, v), written $[\![\pi]\!]_v^G$, are defined by

$$\begin{aligned} [\![id]\!]_v^G &= \{v\}, \\ [\![E_{\text{in}}]\!]_v^G &= \{u \mid (u, v) \in E\}, \\ [\![E_{\text{out}}]\!]_v^G &= \{u \mid (v, u) \in E\}, \\ [\![\top]\!]_v^G &= V. \end{aligned}$$

We consider certain fragments of PML. We define PML_i for $i \in \mathbb{N}$ as the fragment of all PML formulas of modal depth at most i. Let $\mathcal{L}$ be some fragment of PML. We define $\mathcal{L}^{\mathcal{H}}$ as the fragment of all formulas $\varphi \in \mathcal{L}$ where for all modal subformulas the corresponding Peano formula is of the form $\psi = \sum_{i=1}^{K} a_i \cdot \prod_{j=1}^{k} x_{i_j} \le 0$. We define $\mathcal{L}^\top \subseteq \mathcal{L}$ as the fragment of all formulas that only use $\top$ as modality. Likewise, we define $\mathcal{L}^E \subseteq \mathcal{L}$ as the fragment of all formulas that only use E_{in} or E_{out} as modality. We also use combinations such as $\mathcal{L}^{\top,\mathcal{H}}$ in the obvious sense.

Connection between MPNN and PML

Let $\mathcal{M}$ be a class of MPNN, $\mathcal{L}$ a fragment of PML, $\mathcal{G}_\bullet$ a class of pointed graphs, and $c : \mathbb{N} \to (0; 1]$. We say that $M \in \mathcal{M}$ *recognises* φ *over* $\mathcal{G}_\bullet$ *with certainty* $c(|V|)$ if for every $(G = (V, E, L), v) \in \mathcal{G}_\bullet$, it holds that $M(G, v)_\perp = c(|V|)$ when $(G, v) \models \varphi$, and $M(G, v)_\perp = 0$ when $(G, v) \not\models \varphi$. If for each $\varphi \in \mathcal{L}$, there is an $M \in \mathcal{M}$ that recognises it in the above sense for some c dependent of φ, we also denote this by $\mathcal{L} \le \mathcal{M}$ and assume that its clear from context to which pointed graph class it relates. Informally, the notion that M recognises some φ is inspired by binary classification tasks, namely that M aligns with φ up to an input-dependent threshold determined by c.

3 Shallow Peano Modal Logic Fragments

In this section, we focus on analysing the expressive capabilities of MPNN in relation to polynomial counting constraints with shallow modal depth. Formally, we frame this by focusing on fragments of PML_1, which is the fragment of PML where all formulas are of modal depth at most one. Conceptually, we separate our analysis based on the modalities used.

Global Modalities Only

As a first step, we consider the fragment $\text{PML}_1^{\top,\mathcal{H}}$, which includes only modal subformulas exlusively using *homogeneous* Peano terms, which are those where all monomials have the same degree and no constant term. [2]

Lemma 1. *Let* $\varphi = \langle \pi_1, \dots, \pi_m \rangle_\psi(\varphi_1, \dots, \varphi_m) \in \text{PML}_1^{\top,\mathcal{H}}$. *There is* $M_\varphi \in \mathcal{M}_{mean}$ *with* $deg(\psi) + 1$ *layers such that for all graphs* $G = (V, E, L)$ *and* $v \in V$ *we have*

- $(\boldsymbol{x}_v^{(deg(\psi)+1)})_\perp \ge \frac{1}{|V|^{deg(\varphi)}}$ *if* $(G, v) \not\models \varphi$, *and*
- $(\boldsymbol{x}_v^{(deg(\psi)+1)})_\perp = 0$ *otherwise.*

Proof. Let $\chi_1, \dots, \chi_n$ be an enumeration of the subformulas of φ such that

1. if χ_j is a subformula of χ_i then $j \le i$, and
2. $\chi_n = \varphi$.

[2] We remark that Lem. 1 is not a recognition result in the formal sense defined in Section 2, but provides a lower bound on the "certainty" with which an MPNN recognises that a modal formula $\varphi \in \text{PML}_1^{\top,\mathcal{H}}$ is not satisfied by a given pointed graph.

Assume that $deg(\varphi) = k$. We construct $M_\varphi \in \mathcal{M}_{\text{mean}}$ as follows. In the first layer l_1, we evaluate all subformulas χ_i for $i \leq n-1$, such that for all graphs G and nodes v, we have $(\boldsymbol{x}_v^{(1)})_i = 1$ if $(G,v) \models \chi_i$ and $(\boldsymbol{x}_v^{(1)})_i = 0$ if not. Note that these subformulas are exclusively Boolean and, thus, we refer to Lemma 3 for the construction of l_1.

Now, consider the last subformula $\chi_n = \varphi = \langle \pi_1, \ldots, \pi_m \rangle_{\psi(x_1,\ldots,x_m)}(\varphi_1, \ldots, \varphi_m)$. By assumption, we have

- $\psi(x_1, \ldots, x_m) = \sum_{i=1}^K a_i \prod_{j=1}^k x_{i_j} \leq 0$, where $i_j \in \{1, \ldots, m\}$,
- $\pi_i = \top$ for all $i \leq m$.

This indicates that the Peano formula $\psi(x_1, \ldots, x_m)$ needs to be evaluated over values x_{i_j} with $i_j \in \{1, \ldots, m\}$, which represent the total number of nodes in G satisfying the purely Boolean formulas φ_{i_j}. First, we describe how we construct subsequent layers of M_φ to evaluate a single monomial $\prod_{j=1}^k x_{i_j}$ of ψ. We add k layers $l_2, \ldots, l_{2+k-1}$ to M_φ such that $\boldsymbol{x}_v^{(2+(h-1))}$ contains $\frac{1}{|V|^h}\prod_{j=1}^h x_{i_j}$ and $\boldsymbol{x}_v^{(2+(h'-1))}$ contains $x_{i_{h'+1}} \cdot \frac{1}{|V|^{h'}}\prod_{j=1}^{h'} x_{i_j}$ for all $1 \leq h \leq k$ and $1 \leq h' < k$. In the case $h = 1$, we use global mean aggregation of M_φ in layer l_2 over the dimension of the states $\boldsymbol{x}_u^{(1)}$ with $u \in V$ corresponding to φ_{i_1}. This yields $\frac{1}{|V|}x_{i_1}$ in $\boldsymbol{x}_v^{(2)}$ for all $v \in V$, where x_{i_1} represents the number of $u \in V$ such that $G, u \models \varphi_{i_1}$. Then, if $k > 1$, we add the gadget $relu((\frac{1}{|V|}x_{i_1}) - (1 - (\boldsymbol{x}_v^{(1)})_{i_2}))$ to perform the multiplication $(\boldsymbol{x}_v^{(1)})_{i_2} \cdot \frac{1}{|V|}x_{i_1}$ for all $v \in V$. Note that this gadget works as intended due to the fact that $\frac{1}{|V|}x_{i_1} \in [0,1]$ and $(\boldsymbol{x}_v^{(1)})_{i_2} \in \{0,1\}$. For $h > 1$ it works analogously, but we aggregate over the previously computed multiplication value.

Given that we have evaluated the monomial $\prod_{j=1}^k x_{i_j}$ for all $i \leq K$ this way, we can then compute whether $G, v \not\models \varphi$ as stated by the theorem, in layer l_{2+k-1} as follows. Since ψ is homogeneous, the denominator in $\frac{1}{|V|^k}\prod_{j=1}^k x_{i_j}$ is equal for all $i \leq K$. Therefore, we exploit the fact that $\sum_{i=1}^K a_i \prod_{j=1}^k x_{i_j} \leq 0$ if and only if $\frac{1}{|V|^k}\sum_{i=1}^K a_i \prod_{j=1}^k x_{i_j} \leq 0$. Moreover, we note that if the inequality is not satisfied, then $\frac{1}{|V|^k}\sum_{i=1}^K a_i \prod_{j=1}^k x_{i_j} \geq \frac{1}{|V|^k}$. This arises from the fact that $x_{i_j} \in \mathbb{N}$ for all i_j. Thus, we add to $comb_{2+k-1}$ a gadget computing $relu(\sum_{i=0}^K a_i \cdot \frac{1}{|V|^k}\prod_{j=1}^k x_{i_j})$ whose output we make the last dimension of state $\boldsymbol{x}_v^{(2+k-1)}$. $\square$

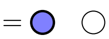
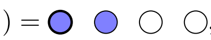
While a restriction, homogeneous polynomial constraints are quite powerful already. Informally, they allow to compare (polynomial) proportions of properties in graphs.

Example 1. *Consider the formula $\varphi = \langle \top, \top, \top \rangle_\psi(p_r, p_g, p_b)$ where $\psi(x_r, x_g, x_b) = x_r^3 - x_g^2 x_b \leq 0$. Clearly, we have $\varphi \in PML_1^{\top,\mathcal{H}}$. Informally, interpret proposition p_r as colour* red, *p_g as colour* green, *and p_b as colour* blue. *Then, the formula φ is satisfied by exactly those graphs G where the cube of the number of red nodes is at most as great as the squared number of green nodes multiplied by the number of blue nodes.*

The obvious next question is whether mean MPNN can recognise all formulas in $\text{PML}_1^\top$. For this purpose, the arguments used in Lemma 1 prove to be insufficient since the denominators $|V|^i$ introduced while evaluating monomials are no longer guaranteed to be identical, as the degree i of monomials in Peano terms may vary. Moreover, potential constant terms in Peano expressions also need to be considered. It is, in fact, straightforward to demonstrate that for arbitrary inputs, mean MPNN are unable to express all such constraints without further restrictions.

Lemma 2. *Let $\varphi = \langle \top \rangle_{x_1 \geq 2}(p) \in PML_1^\top$. There is no $M \in \mathcal{M}_{mean}$ such that for all pointed graphs (G,v) we have M accepts (G,v) if and only if $(G,v) \models \varphi$.*

Proof. Assume the contrary, namely that there exists an $M \in \mathcal{M}_{\text{mean}}$ that recognises φ exactly. W.l.o.g. assume that p corresponds to the label dimension 0. Consider the two pointed graphs $(G_1 = (V_1, E_1, L_1), v_1)$ with $V_1 = \{v_1, v_2\}$, $E_1 = \emptyset$, $L_1(v_1) = (1)$, $L_1(v_2) = (0)$, and $(G_2 = (V_2, E_2, L_2), v_1')$ with $V_2 = \{v_1', v_2', v_3', v_4'\}$, $E_2 = \emptyset$, $L_2(v_i') = (1)$ if $i \leq 2$, and $L_2(v_i') = (0)$ if $i \geq 3$. Visually, we have

$(G_1, v_1) =$  and $(G_2, v_1') =$ ,

where the blue filling denotes label (1), white filling denotes label (0), and the focus is thickly bordered. It is easy to see that $(G_1, v_1) \not\models \varphi$ and $(G_2, v_1') \models \varphi$. However, it is also straightforward that for each pair of nodes $(u, u') \in V_1 \times V_2$ with $L_1(u) = L_2(u')$ that all applications of global mean aggregations lead to the same values. Given that, we have that $\boldsymbol{x}_{v_1}^{(k)} = \boldsymbol{x}_{v_1'}^{(k)}$, where k is the last layer of M, this contradicts the fact that M does not accept (G_1, v_1) but (G_2, v_1'). $\square$

We identify a simple assumption about input graphs, enabling mean MPNN to recognise general polynomial constraints: there is a marked node, meaning one node that has a colour not shared by any other node.

Theorem 1. *Let $\varphi \in PML_1^\top$. There is $M_\varphi \in \mathcal{M}_{mean}$ with $\mathcal{O}(deg(\varphi))$ layers that recognises φ over $(\mathcal{G})^p_\bullet$ with certainty $\frac{1}{|V|^{\mathcal{O}(deg(\varphi))}}$.*

Proof. The proof works similar to Lemma 1, namely we give a construction for M_φ. Again, we enumerate the subformulas of φ such that if χ_j is a subformula of χ_i, then $j \leq i$, with $\chi_n = \varphi$. Additionally, we ensure that there is some $h \leq n$ such that for all χ_j we have $md(\chi_j) = 0$ if $j \leq h$ and $md(\chi_j) = 1$ if $j > h$. In other words, the enumeration starts with all formulas that do not contain a modal subformula. Let $G = (V, E, L) \in \mathcal{G}^p$ be a graph with marked node $u \in V$. We construct M_φ as follows. Layer l_1 is constructed as in Lemma 3, evaluating all formulas χ_i with $i \leq h$.

Consider $\chi_i = \langle \pi_1, \ldots, \pi_m \rangle_{\psi(x_1,\ldots,x_m)}(\varphi_1, \ldots, \varphi_m)$. By assumption, we know that $md(\chi_i) = 1$. In contrast to the setting of Lemma 1, there are three differences regarding the Peano formula ψ:

1. monomials of a term $\sum_{i=1}^{K} a_i \cdot \prod_{j=1}^{k_i} x_{i_j} \leq b$ are of the form $\prod_{j=1}^{k_i} x_{i_j}$, indicating they may have varying degrees k_i,
2. terms may include a constant part $b \neq 0$, and
3. ψ can be a genuine Peano formula, meaning its a Boolean combination of Peano terms.

We continue by describing how to evaluate monomials and constant parts. Let $deg(\psi) = \hat{k} = \max(k_1, \ldots, k_m)$ be the maximum degree of all monomials occurring in ψ. We add layers $l_2, \ldots, l_{2+\hat{k}-1}$ so that they perform the same kind of computation as in Lemma 1. This yields that $\boldsymbol{x}_v^{(2+(k_i-1))}$ contains $y_v = \frac{1}{|V|^{k_i}} \prod_{j=1}^{k_i} x_{i_j}$ for all $i \leq m$ and $v \in V$. We utilise that there is a marked node u to align the different denominators introduced while evaluating monomials. Specifically, in layer l_{2+k_i-1}, we add a gadget to $comb_{2+k_i-1}$ that computes $z_v = relu(y_v - (1 - (\boldsymbol{x}_v^{(2+k_i-2)})_{mark}))$, where $(\boldsymbol{x}_v^{(2+k_i-2)})_{mark}$ denotes the dimension identifying the marked node u. We remark that it is no longer guaranteed to be dimension p due to pontential shifts in the dimensionality of the states. Note that it is guaranteed that $y_v \in [0,1]$ for all $v \in V$ and, thus, $z_v = y_v$ if $u = v$ and $z_v = 0$ otherwise. Now, using mean global aggregation in layer l_{2+k_i} over the dimension corresponding to z_v computes $\frac{1}{|V|^{k_i+1}} \prod_{j=1}^{k_i} x_{i_j}$. By applying this construction repeatedly, we enable M_φ to align all denominators to $|V|^{\hat{k}}$.

Similarly, we exploit the presence of a marked node u and add the gadget $(\boldsymbol{x}_v^{(2)})_b = relu(|b| \cdot (\boldsymbol{x}_v^{(1)})_{mark})$ to $comb_2$. Note, that in the final evaluation of the corresponding Peano term, we multiply this value by $\operatorname{sgn}(b)$. Then, using global aggregation $\hat{k}$ times over the dimension $(\boldsymbol{x}_v^{(1+i)})_b$ computes the value $\frac{b}{|V|^{\hat{k}}}$. Note that this involves using gadget $relu((\boldsymbol{x}_v^{(1+i)})_b - relu(b - b(\boldsymbol{x}_v^{(1+i)})_{mark}))$ to store a nonzero value if and only if $v = u$. Given that we evaluated all monomials and constants that way, we refer to Lemma 4 (with $r_1 = r_2 = \frac{1}{|V|^{\hat{k}}}$) stating how to evaluate ψ and thereby χ_i in $comb_{2+\hat{k}-1}$ such that $\boldsymbol{x}_v^{(2+\hat{k}-1)} = \frac{1}{|V|^{\hat{k}}}$ if $(G,v) \models \chi_i$ and $\boldsymbol{x}_v^{(2+\hat{k}-1)} = 0$ if $(G,v) \not\models \chi_i$.

Finally, the remaining subformulas χ_i, which are arbitrary Boolean formulas from PML_1, are handled in $l_{2+deg(\varphi)-1}$ as follows. First, we adjust the value of χ_j with $j \leq h$ using gadget $relu(\frac{1}{|V|^{\hat{k}}} - relu(1 - x_{\chi_j}))$, where x_{χ_j} represents the previous evaluation of χ_j. This ensures that the semantics of all χ_j with $j < i$ are represented by 0 and $\frac{1}{|V|^{\hat{k}}}$. Second, we add gadgets $\neg\chi_1 = relu(\frac{1}{|V|^{\hat{k}}} - x_{\chi_1})$ and $\chi_1 \wedge \chi_2 = relu(x_{\chi_1} + x_{\chi_2} - \frac{1}{|V|^{\hat{k}}})$ and combine them as determined by the structure of χ_i. Note that the value $\frac{1}{|V|^{\hat{k}}}$ can be computed by repeated global mean aggregations, analogously to how we adjusted the bias value b. $\square$

Local Modalities Only

While shallow PML formulas involving only global modalities are capable of expressing interesting global properties, they cannot convey information about local neighbourhoods. Thus, we turn our attention to PML_1^E next.

Example 2. *Consider the formula* $\varphi = \langle E_{in}, E_{out} \rangle_\psi (p_b, p_g \wedge \neg p_r)$, *where* $\psi(x_b, x_{g\wedge\neg r}) = x_b^2 - x_{g\wedge\neg r}^3 \leq 1$, *and interpret p_b as the colour* blue*, p_g as the colour* green *and p_r as the colour* red*. The formula φ is satisfied by all pointed graphs (G, v) where the squared number of ingoing neighbours of colour blue of v is at most as large as the cube of the number of outgoing neighbours of colour green but not red plus one of v.*

In contrast to the settings of Lemma 1 and Theorem 1, capturing formulas from PML_1^E with mean MPNN obviously requires the use of local aggregation. However, local mean aggregation generally does not exhibit uniformity since each node's aggregated value depends on the size of its neighbourhood. We identify that assuming the following conditions enables MPNN from $\mathcal{M}_{\text{mean}}$ to recognise said formulas: (a) the graphs (G, v) are regular, (b) v is marked, (c) v has a self-loop.

Theorem 2. *Let $\varphi \in \text{PML}_1^E$. There is $M_\varphi \in \mathcal{M}_{mean}$ with $\mathcal{O}(deg(\varphi))$ layers that recognises φ over all regular pointed graphs in $(G, v) \in (\mathcal{G})^p_\bullet$ where v is strongly marked with certainty $\frac{1}{|V|^{\mathcal{O}(deg(\varphi))}}$.*

Proof. We enumerate the subformulas of φ as a sequence $\chi_1, \ldots, \chi_n$ such that if χ_j is a subformula of χ_i, then $j \leq i$ holds, there is some $h \leq n$ such that $md(\chi_j) = 0$ if and only if $j \leq h$, and $\chi_n = \varphi$. Let $(G = (V, E, L), v) \in (\mathcal{G})^p_\bullet$ be a regular pointed graph where v is marked and $v \in \text{neigh}(v)$. Since G is regular, we denote by n_{in} the size of all ingoing and by n_{out} the size of all outgoing neighbourhoods. We construct M_φ as follows. Layer l_1 is constructed as in Lemma 3 in order to evaluate all χ_i with $i \leq h$.

Consider $\chi_i = \langle \pi_1, \ldots, \pi_m \rangle_{\psi(x_1,\ldots,x_m)}(\varphi_1, \ldots, \varphi_m)$. By assumption, we know that $md(\chi_i) = 1$ and all $\pi_i \in \{E_{\text{in}}, E_{\text{out}}\}$, implying that M_φ has to evaluate ψ at node $v \in V$ over values x_{i_j} equal to the number of nodes $u \in \text{neigh}_{\text{in}}(v)$ or $u \in \text{neigh}_{\text{out}}(v)$, depending on the form of π_{i_j}, with $(G, u) \models \varphi_{i_j}$. We add layers $l_2, \ldots, l_{2k_i}$ to M_φ to evaluate monomials $\prod_{j=1}^{k_i} x_{i_j}$ of χ_i in the following way: W.l.o.g. assume that $\pi_{i_1} = E_{\text{in}}$. In layer l_2 we first compute $y_u = \frac{1}{n_{\text{in}}} x_{i_1}$ for each node $u \in V$ using loc_{in}, similarly to our constructions in Lemma 1 or Theorem 1. Then, we add the gadget $z_u = relu(y_u - (1 - (\boldsymbol{x}_u^{(1)})_{mark}))$ to $comb_2$ where $mark$ corresponds to the state dimension that marks the focus v (not necessarily equal to p anymore due to potential shifts in layer l_1). Then, $z_u \neq 0$ at node u if and only if $u = v$. W.l.o.g. assume that $\pi_{i_2} = E_{\text{out}}$. In layer l_3, we use loc_{in} and add to $comb_3$ a gadget which in combination computes $y'_u = \mathbf{1}_{G,u \models \varphi_{i_2}} \cdot \frac{1}{n_{\text{in}}} \sum_{w \in \text{neigh}(u)} z_w$ for all $u \in V$. The respective gadget is identical to the one used in previous constructions to perform multiplication. Now, y'_u is only non-zero for $u \in \text{neigh}(v)$, yielding $y'_u = \mathbf{1}_{G,u \models \varphi_{i_2}} \cdot \frac{1}{n_{\text{in}}} y_v$. Next, in layer l_4 we use loc_{out}, and compute $\frac{1}{n_{\text{in}}(n_{\text{in}} n_{\text{out}})} \prod_{j=1}^{2} x_{i_j}$ at node v. Using this construction repeatedly results in the value $\frac{1}{n_{\text{in}}(n_{\text{in}} n_{\text{out}})^{k_i - 1}} \prod_{j=1}^{k_i} x_{i_j}$ in layer l_{2k_i}.

To align necessarily introduced denominators for all monomial evaluations, we perform repeated ingoing or outgoing local aggregations until each denominator is $\frac{1}{(n_{\text{in}} n_{\text{out}})^{deg(\chi_i)}}$. Note that this requires adding $relu(x - relu(1 - x_{\text{mark}}))$ gadgets to ensure that non-zero values occur only at v. Similarly, we adjust layers $l_2, \dots, l_{2deg\varphi+1}$ to compute $\frac{1}{(n_{\text{in}} n_{\text{out}})^{deg(\varphi)}} b$ for all constant parts b ocurring in ψ. From here onward, we proceed as stated in Lemma 4 to evaluate χ_i using $r_1 = \frac{1}{(n_{\text{in}} n_{\text{out}})^{deg(\varphi)}}$ and $r_2 = \frac{1}{|V|^{2deg(\varphi)}}$, which we can compute using repeated global aggregation in in layers l_1 to $l_{2deg(\varphi)}$.

The remaining construction of M_φ is exactly as described in the proof of Theorem 1. □

We remark that if we restrict to homogeneous constraints, assumption (c), meaning that the focus v has a self-loop, could be dropped.

The limitation to regular graphs may restrict the applicability in most scenarios. However, allowing for the use of sum or max aggregation functions alongside mean aggregations enables us to show that such MPNN can recognise PML_1^E over all graphs that satisfy properties (b) and (c) as previously discussed.

Theorem 3. *Let $\varphi \in PML_1^E$ and $A = \{mean, sum\}$ or $A = \{mean, max\}$. There is $M_\varphi \in \mathcal{M}_A$ with $\mathcal{O}(deg(\varphi))$ layers that recognises φ over all graphs $(G, v) \in \mathcal{G}_\bullet^p$ where v is strongly marked with certainty $\frac{1}{|V|^{\mathcal{O}(deg(\varphi))}}$.*

Proof. The construction of M_φ works analogous to Theorem 2. However, while evaluating monomials $\prod_{j=1}^{k_i} x_{i_j}$, we alternate between realisations of loc_{out} and loc_{in} either by mean or by sum (or max) aggregation.

As done in the proof of Theorem 2, we employ local mean aggregation, w.l.o.g assume its loc_{in}, and a specific gadget in l_2 to compute $z_u = \frac{1}{|\text{neigh}_{\text{in}}(v)|} x_{i_1}$ if $u = v$, and $z_u = 0$ otherwise. In layer l_3, instead of mean aggregation, we use outgoing local sum aggregation (or max aggregation) to compute $y'_u = \mathbf{1}_{G,u \models \varphi_{i_2}} \cdot \sum_{w \in \text{neigh}(u)} z_w$ or $y'_u = \mathbf{1}_{G,u \models \varphi_{i_2}} \cdot \max_{w \in \text{neigh}(u)} z_w$ for all $u \in V$. Subsequently, in l_4, we revert to local mean aggregation and continue as before. This construction introduces a denominator of the form $\frac{1}{n_{i_1} \cdot \dots \cdot n_{i_{deg(\varphi)}}}$ where $n_{i_j} \in \{|\text{neigh}_{\text{in}}(v)|, |\text{neigh}_{\text{out}}(v)|\}$ to each evaluated monomial. However, in order to align these denominators across all monomials and constants leads to $\frac{1}{|\text{neigh}_{\text{in}}(v)|^{deg(\varphi)} \cdot |\text{neigh}_{\text{out}}(v)|^{deg(\varphi)}} \geq \frac{1}{|V|^{2deg(\varphi)}}$ in the worst case.

Aside from these adjustments, the construction proceeds exactly as in Theorem 2. □

Mixed Modalities

In the settings of Theorem 1 or Theorem 2 and 3, we assumed that modal formulas use uniform modalities, in the sense that they are either $\top$ or $E_{\text{in}}, E_{\text{out}}$, but not both simultaneously. Furthermore, we did not yet consider the modality id. However, combinations of these modalities bring the full expressive power of PML_1.

Example 3. *Let $\varphi = \langle E_{in}, \top, id \rangle (p_r \wedge p_b, p_b, p_g \vee p_y)$, where we use $\vee$ as the usual abbreviation for logical disjunction, and $\psi(x_{r\wedge b}, x_b, x_{g\vee y}) = (16 \leq x_{r\wedge b}^2 + x_b x_{g\vee y}) \wedge (x_b^3 + x_{r\wedge b}(1 - x_{g\vee y}) \leq 64)$. As in previous examples, we interpret p_r as colour* red*, p_b as colour* blue*, p_g as colour* green*, and p_y as colour* yellow*. Informally, the formula φ is satisfied by exactly those pointed graphs (G, v) where*

- *if v is of colour green or yellow, we have that the square of the number of ingoing neighbours of v which are red and blue plus the global number of blue nodes is at least 16 ($16 \leq x_{r\wedge b}^2 + x_b \cdot 1$), and the cube of the number of blue nodes in G is at most 64 ($x_b^3 + x_{r\wedge b} \cdot 0 \leq 64$); or*
- *if v is neither of colour green nor yellow, we have that the square of the number of ingoing neighbours of v which are red and blue is at least 16 ($16 \leq x_{r\wedge b}^2 + x_b \cdot 0$), and the cube of the number of blue nodes in G plus the number of ingoing neighbours of v that are red and blue is at most 64 ($x_b^3 + x_{r\wedge b} \cdot 1 \leq 64$).*

As it turns out, in order to capture all formulas from PML_1, it suffices to combine the constructions developed in previous sections.

Theorem 4. *Let $\varphi \in PML_1$. There is $M_\varphi \in \mathcal{M}_{mean}$ with $\mathcal{O}(deg(\varphi))$ layers that recognises φ over all regular graphs in $(G, v) \in (\mathcal{G})_\bullet^p$ where v is strongly marked with certainty $\frac{1}{|V|^{\mathcal{O}(\deg(\varphi))}}$.*

Proof. The proof follows the same line of arguments as used in the proofs of Theorem 1 and Theorem 2.

To allow M_φ to evaluate monomials $\prod_{j=1}^{k_i} x_{i_j}$ for a given regular pointed graph $(G = (V, E, L), v)$, where x_{i_j} corresponds to a modality $\pi_{i_j} \in \{\text{id}, E_{\text{in}}, E_{\text{out}}, \top\}$, we adjust and combine previous constructions as follows. If x_{i_j} corresponds to $\top$, we evaluate it as described in Theorem 1 using global mean aggregation, thus introducing denominators of the form $\frac{1}{|V|^i}$. If x_{i_j} corresponds to E_{in} or E_{out}, we evaluate it as described in Theorem 2, which introduces denominators of the form $\frac{1}{(n_{\text{in}} n_{\text{out}})^i}$, where n_{in} and n_{out} are the sizes of ingoing and outgoing neighbourhoods of the regular graph G. If x_{i_j} corresponds to id, meaning aggregation over the focus node v only, we use the fact that v is marked, and (i) adjust all dimensions x of interest by $relu(x - relu(1 - x_{\text{mark}}))$, where x_{mark} corresponds to the dimension marking v, storing the result in new dimensions; (ii) use global or local aggregation. Note that depending on the choice this introduces $\frac{1}{|V|^i}$, $\frac{1}{n_{\text{in}}^i}$, or $\frac{1}{n_{\text{out}}^i}$. Finally, we use repeated global, ingoing, or outgoing aggregation to align all denominators introduced in the values representing monomials and constants. We remark that this requires a linear amount of layers regarding $deg(\varphi)$, depending on distribution of different modalities across monomials.

Otherwise, the construction remains the same as in previous proofs. $\square$

Allowing for either sum or max aggregations besides mean aggregations allows to drop the regularity assumption. The proof requires the exact same adjustments as made in the step from Theorem 2 to Theorem 3.

Theorem 5. *Let $\varphi \in PML_1$ and $A = \{mean, sum\}$ or $A = \{mean, max\}$. There is $M_\varphi \in \mathcal{M}_A$ with $\mathcal{O}(deg(\varphi))$ layers that recognises φ over all graphs in $(G, v) \in (\mathcal{G})^p_\bullet$ where v is strongly marked with certainty $\frac{1}{|V|^{\mathcal{O}(deg(\varphi))}}$.*

4 Nested Peano Modal Logic Fragments

In this section, we consider fragments of PML of arbitrary modal depth. Contrary to the settings of Lemma 1 and Theorem 1, focusing on fragments with nested modal formulas but using only $\top$ as a modality does not bring any new insights into the power of mean MPNN since nesting does not enhance the expressive capabilities of these fragments compared to $\text{PML}_1^\top$, as demonstrated in the following.

Proposition 1. *For all $\varphi \in PML^\top$ there is $\varphi' \in PML_1^\top$ with $[\![\varphi]\!] = [\![\varphi']\!]$.*

Proof. Let φ be a PML formula with only global modalities. For a set T of modal formulas and a PML formula γ, we write γ^T for the formula where every strict modal subformula μ of γ is replaced with $\top$ if $\mu \in T$ and with $\bot$ otherwise. Here, strict means that if γ is a modal formula, then γ itself is neither replaced with $\top$ nor $\bot$. Let S be the set of strict modal subformulas of φ. Now φ is equivalent to

$$\bigvee_{T \subseteq S} \left(\varphi^T \wedge \bigwedge_{\tau \in T} \tau^T \wedge \bigwedge_{\sigma \in S \setminus T} \neg \sigma^T\right)$$

which is a formula of modal depth at most 1. Here, the modal subformulas can be checked without nesting them since they only have global modalities, i.e., all modal formulas are evaluated over the same pointed graph. $\square$

However, the situation is entirely different if we consider local modalities $E_{\text{in}}, E_{\text{out}}$ only. Let $\varphi \in \text{PML}^E$ and $(G = (V, E, L), v)$ be a pointed graph. We denote by $Trace^{\leq i}_\varphi(u)$ the set of all traces $E_0, \ldots, E_{j-1} \in Trace^j_\varphi$ with $j \leq i$ such that there is a walk from v to u respecting $E_0, \ldots, E_{j-1}$. We define the class $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ of all pointed graphs $(G = (V, E, L), v)$ where v is marked and for all $w \in V$ and $1 \leq i \leq md(\varphi)$ if there is a walk $u_0 u_1 \cdots u_{i-1} w$ with $u_0 = v$ that corresponds to a trace $E_0, \ldots, E_{i-1} \in Trace^i_\varphi$ then we have $u_{i-1} \in \text{neigh}(u_{i-1})$ and for all $w_1, w_2 \in \text{neigh}_x(w)$ with $x = \text{in}$ if $E_{i-1} = E_{\text{out}}$ and $x = \text{out}$ if $E_{i-1} = E_{\text{in}}$ we have that if $Trace^{\leq i-1}_\varphi(w_1) = Trace^{\leq i-1}_\varphi(w_2)$ then $w_1 = w_2$.

While the definition of $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ is rather technical, assuming for instance that $Mod^i_\varphi = \{E_{\text{out}}\}$ for all $i \leq md(\varphi)$ implies that $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ includes trees with self-loops where the focus is the root. In Figure 2, we depict some examples. We identify $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ as a suitable candidate to frame the expressive capabilities of mean MPNN compared to PML^E.

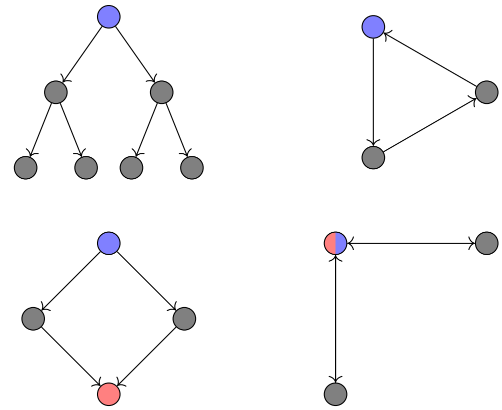


Figure 2: Let φ be a formula with $md(\varphi) = 2$ that only uses the modality E_{out}. The top two graphs are members of $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$, where the blue node indicates the focus. The bottom graphs are not included in $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ since multiple predecessors of the red node have the same trace set. For clarity reasons, self-loops are not depicted.

Theorem 6. *Let $\varphi \in PML^E$. There is $M_\varphi \in \mathcal{M}_{mean}$ with $\mathcal{O}(md(\varphi)deg(\varphi))$ layers that recognises φ over all regular graphs in $(\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$ with certainty $\frac{1}{|V|^{\mathcal{O}(md(\varphi)deg(\varphi))}}$.*

Proof. Let $\chi_1, \ldots, \chi_n$ be an enumeration of the subformulas of φ such that if χ_j is a subformula of χ_i, then $j \leq i$, there is some $h \leq n$ such $md(\chi_i) = 0$ if and only if $i \leq h$ for all $i \leq n$, and we have $\chi_n = \varphi$. W.l.o.g. we assume that $p = 0$. Let $(G = (V, E, L), v) \in (\mathcal{T}^{\circlearrowleft}_\varphi)^p_\bullet$. Similar to previous proofs, we describe how to construct M_φ. Layer l_1 is constructed as described by Lemma 3 and used to evaluate all χ_i with $i \leq h$.

Next, we use layers $l_1, \ldots, l_{md(\varphi)-1}$ to "propagate" the mark of the focus throughout the graph along traces of $Trace^i_\varphi$ for $i \leq md(\varphi) - 1$. In layer l_i we use loc_{in} if there is trace $T = E_0, \ldots, E_{i-1} \in Trace^i_\varphi$ with $E_{i-1} = E_{\text{out}}$ and loc_{out} if there is trace $T = E_0, \ldots, E_{i-1} \in Trace^i_\varphi$ with $E_{i-1} = E_{\text{in}}$ to aggregate for all $u \in V$ a new flag y_T stored in some dimension associated with T. Simultaneously, we use $glob$ in each layer l_i to compute $z_i = \frac{1}{|V|^i}$ through repeated aggregation over the marking dimension as done in previous constructions. Finally, in $comb_i$, we compute $\min(z_i, y_T)$ for each $T \in Trace^i_\varphi$. This ensures that we stored $\frac{1}{|V|^i}$ in a dimension associated with T if there exists a walk from v to u respecting $T \in Trace^i_\varphi$, and its 0 otherwise.

Next, in layer $l_{md(\varphi)}$, we employ the value $\frac{1}{|V|^{md(\varphi)-1}}$, obtained through repeated global aggregations in layers $l_1, \ldots, l_{md(\varphi)-1}$, to compute $(\boldsymbol{x}^{(md(\varphi))}_u)_i = relu(\frac{1}{|V|^{md(\varphi)-1}} - relu(1 - (\boldsymbol{x}^{(md(\varphi)-1)}_u)_i))$ for all $u \in V$ and $i \leq h$. Informally, this means a shift of the representative values for all so far evaluated formulas (which are exclusively Boolean) with $0 \mapsto 0$ and $1 \mapsto \frac{1}{|V|^{md(\varphi)-1}}$.

Now, consider subformula $\chi_i =$

$\langle \pi_{i,1}, \ldots, \pi_{i,m} \rangle_{\psi_i(x_{i,1},\ldots,x_{i,m})}(\varphi_{i,1}, \ldots, \varphi_{i,m})$ with $md(\chi_i) = 1$. First, we argue how to evaluate monomials $\prod_{j=1}^{k_i} x_{i_j}$ in ψ_i. W.l.o.g assume that the modal depth of χ_i implies that χ_i must be evaluated at nodes u such that there exists a walk starting from v to u respecting some $T \in \mathit{Trace}_{\varphi}^{md(\varphi)-1}$. We remark that the structure of φ exactly characterises at which distance from v some χ_i must be evaluated. We fix such a node u in the following. We utilise layers $l_{md(\varphi)+1}, \ldots, l_{md(\varphi)+2deg(\varphi)-1}$ to evaluate $\prod_{j=1}^{k_i} x_{i_j}$ as done in Theorem 2, but with the following adjustments. W.l.o.g. assume that x_{i_1} corresponds to modality E_{out} and x_{i_2} corresponds to E_{in}. We use $\mathit{loc}_{\text{out}}$ in layer $l_{md(\varphi)+1}$ as before. Now, in $\mathit{comb}_{md(\varphi)+1}$ we add gadget $\mathit{relu}(x - \sum_{T \in I}(\frac{1}{|V|^{|T|}} - y_T) - \sum_{T \in \mathit{Trace}_{\varphi}^{\leq md(\varphi)-1} \setminus I} y_T)$ for all $I \subseteq \mathit{Trace}_{\varphi}^{\leq md(\varphi)-1}$ where y_T is the flag we previously computed with $y_T = \frac{1}{|V|^{|T|}}$ if there is a walk from v to u respecting T. This ensures that we preserve a non-zero value corresponding to the partial evaluation of $\prod_{j=1}^{k_i} x_{i_j}$ at u in a dimension identified by $\mathit{Trace}_{\varphi}^{\leq md(\varphi)-1}(u)$. Note that in order for this gadget to work correctly we require the shift in values (0 to 0 and 1 to $\frac{1}{|V|^{md(\varphi)-1}}$) which is performed by layer $l_{md(\varphi)}$. Due to the definition of $(\mathcal{T}_{\varphi}^{\circlearrowleft})_{\bullet}^{p}$ it is ensured that all nodes $w \in \text{neigh}_{\text{in}}(u)$ have at most one $u' \in \text{neigh}_{\text{out}}(w)$ with a specific trace set $\mathit{Trace}_{\varphi}^{\leq md(\varphi)-1}(u')$. Thus, the evaluation of $\prod_{j=1}^{k_i} x_{i_j}$ proceeds as before, but separately for each dimension corresponding to some $\mathit{Trace}_{\varphi}^{\leq md(\varphi)-1}(u')$ with $u' \in V$. In the end, we utilise that u has a self-loop, again as given by the definition of $(\mathcal{T}_{\varphi}^{\circlearrowleft})_{\bullet}^{p}$, to align the denominators for all monomials by repeated local aggregations along this self-loop. Note that this only works due to G being regular, similar to Theorem 2. Given that we evaluated all monomials in ψ_i this way and normalised constant parts of ψ_i in the same way, we evaluate χ_i as described in Lemma 4 with $r_2 = \frac{1}{|V|^{md(\varphi)-1+2deg(\varphi)}}$, computed using repeated global aggregations in previous layers.

Now, for modal formulas χ_i with $md(\chi_i) > 1$ we proceed analogously, but we adjust the values representing all so far evaluated formulas to $\frac{1}{|V|^{md(\varphi)+((md(\chi_i)-1)2deg(\varphi))}}$ and 0, assuming that χ_i must be evaluated at nodes u with distance $md(\varphi) - md(\chi_i)$ from v. For all χ_j with $j \leq h$ this works as described before in preparation of handling $md(\chi_i) = 1$ subformulas. For χ_j which are already evaluated modal formulas we take the minimum of the currently representing value and $\frac{1}{|V|^{md(\varphi)+((md(\chi_i)-1)2deg(\varphi))}}$. Then, after resolving all modal formulas with $md(\chi_i) = md(\varphi)$ we evaluate the remaining subformulas as in Theorem 2, but with $\frac{1}{|V|^{md(\varphi)(2deg(\varphi))}}$. Note that this, again, requires shifting representative values. $\square$

Analogous to Theorem 3, we get that also allowing for sum or max as aggregation besides taking the mean in turn allows us to drop the regularity assumption.

Theorem 7. *Let $\varphi \in \mathit{PML}^E$, and $A \in \{\{\mathit{mean}, \mathit{sum}\}, \{\mathit{mean}, \mathit{max}\}\}$. There is $M_\varphi \in \mathcal{M}_A$ that recognises φ over all graphs in $(\mathcal{T}_{\varphi}^{\circlearrowleft})_{\bullet}^{p}$ with certainty $\frac{1}{|V|^{\mathcal{O}(md(\varphi)deg(\varphi))}}$.*

5 Summary and Outlook

We investigated the capabilities of MPNN, primarily using mean aggregation, in recognising polynomial count properties in graphs. We demonstrated that, without further assumptions, mean MPNN that are capable of capturing homogeneous polynomial constraints over global properties (Lemma 1). Subsequently, we proved that a simple assumption about pointed graphs, namely that the focus is marked, allows mean MPNN to recognise arbitrary polynomial constraints over regular pointed graphs, counting globally or locally without nesting (Theorem 1 and Theorem 2). Interestingly, we found that by allowing additional sum or max aggregations, the regularity assumption can be omitted (Theorem 5). Considering nested modalities, we demonstrated that for graphs with a somewhat tree-like structure, the mean plus sum or max MPNN can capture all polynomial constraints (Theorem 7). Again, when considering MPNN using only mean aggregation, we required a restriction to regular, tree-like graphs (Theorem 6).

Building on our findings, future work should explore to what extent we can generalise PML along the modality dimension. For example, whether allowing arbitrary Boolean combinations of the atomic modalities id, E_{in}, and E_{out} leads to similar findings. Additionally, it can be envisaged that MPNN are also capable of comparing counts of neighbourhoods at different distances to the focus point. However, this is not incorporated into the logic PML and, thus, requires additional work.

Acknowledgment. We thank anonymous reviewers for their helpful feedback. The authors are partially supported by Deutsche Forschungsgemeinschaft (grant number 522843867) and European Union[3] (ERC, LASD, 101089343).

AI Declaration

The authors have not employed any Generative AI tools.

A Omitted Technicalities

In the following technical lemmas and statements, we always assume that comb functions are realised by FNN with ReLU ($\mathit{relu}(x) = \max(0, x)$) activations.

We denote by PL the fragment of PML formulas φ such that there is no modal formula $\varphi' = \langle \pi_1, \ldots, \pi_m \rangle_{\psi(x_1,\ldots,x_m)}(\varphi_1, \ldots, \varphi_m)$ with $\varphi' \in \mathit{sub}(\varphi)$. In other words, PL are effectively propositional formulas.

[3]Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Lemma 3. *Let $\varphi_1, \ldots, \varphi_n \in PL$ over the finite set of propositions P. There is a combination function comb : $\{0,1\}^{4|P|} \to \{0,1\}^n \times \{0\}^m$ for some $m \in \mathbb{N}$, such that for all graphs $G = (V, E, L)$ with $|P|$ colours, nodes $v \in V$, and $i \in \{1, \ldots, n\}$ we have*

$$(comb(\boldsymbol{x}_v^{(0)}, \boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3))_i = 1 \quad \text{iff} \quad G, v \models \varphi_i$$

where $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, and $\boldsymbol{y}_3$ are the results coming from any form of local and global aggregations.

Proof. Due to the fact that all formulas $\varphi_1, \ldots, \varphi_n \in$ PL, implying that we do not need to use any information coming from local or global aggregation to evaluate these. Therefore, we construct the FNN N representing $comb$ such that it simply ignores the inputs $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, and $\boldsymbol{y}_3$. This is done by weighting these inputs with 0 in the first layer of N.

Let $G = (V, E, L)$ be a graph with $|P|$ colours and $v \in V$. We proceed the construction of N as follows. If $\chi_i = p_j$, we add the gadget $relu((\boldsymbol{x}_v^{(0)})_j)$ to N. In other cases, we use $relu(1 - x_j)$ if $\chi_i = \neg\chi_j$, and $relu(x_{j_1} + x_{j_2} - 1)$ if $\chi_i = \chi_{j_1} \wedge \chi_{j_2}$. Here, the values x_j, x_{j_1} and x_{j_2} may not necessarily correspond to dimensions of $\boldsymbol{x}_v^{(0)}$, but can also correspond to the outputs of previously applied gadgets. □

Let $\psi(x_1, \ldots, x_n)$ be a Peano formula. We denote by Mon_ψ the set of all monomials $\prod_{j=1}^{k_i} x_{i_j}$ occurring in terms of ψ and $Const_\psi$ the set of all constants b occurring in terms of ψ.

Lemma 4. *Let $\psi(x_1, \ldots, x_n)$ be a Peano formula with $Mon_\psi = \{M_1, \ldots, M_m\}$ and $Const_\psi = \{b_1, \ldots, b_k\}$, and $(c_1, \ldots, c_n) \in \mathbb{N}^n$ an assignment for ψ. For all $r_1, r_2 \in \mathbb{Q}$ with $r_1 \geq r_2 \geq 0$ there is combination function comb such that*

$$(comb(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2, \boldsymbol{y}_3))_i = r_2 \quad \text{iff} \quad (c_1, \ldots, c_n) \models \psi,$$

where $\boldsymbol{x}$ contains the values $r_1 \cdot M_i(c_1, \ldots, c_n)$, $r_1 \cdot b_j$, r_2, for all $i \leq m$ and $j \leq k$, and $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, and $\boldsymbol{y}_3$ are the results coming from any form of local and global aggregations.

Proof. Similar to Lemma 1, whether a Peano term $t = \sum_{i=1}^{K} a_i \cdot \prod_{j=1}^{k_i} x_{i_j} \leq b$ is satisfied by $(c_1, \ldots, c_n)$ is checked using the gadget $x_t = relu((r_1 \sum_{i=1}^{K} a_i \cdot \prod_{j=1}^{k_i} x_{i_j}) - r_1 b)$ in the sense that $x_t = 0$ if $(c_1, \ldots, c_n) \models t$ and $x_t \geq r_1$ if not. Next, we use gadget $y_t = relu(x_t - relu(x_t - r_2))$ to compute $y_t = \min(x_t, r_2)$. Finally, we use $z_t = relu(r_2 - y_t)$ to get that $z_t = 0$ if $(c_1, \ldots, c_n) \not\models t$ and $z_t = r_2$ if $(c_1, \ldots, c_n) \models t$.

To fully evaluate ψ, which potentially is a Boolean combination of Peano terms, we we handle Boolean combinations using gadgets $\neg\psi_1 = relu(r_2 - x_{\psi'})$ and $\psi_1 \wedge \psi_2 = relu(x_{\psi_1} + x_{\psi_2} - r_2)$ and combine them as determined by the structure of ψ. □

References

Abboud, R.; Ceylan, İ. İ.; Grohe, M.; and Lukasiewicz, T. 2021. The surprising power of graph neural networks with random node initialization. In Zhou, Z., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, 2112–2118. ijcai.org.

Ahvonen, V.; Heiman, D.; Kuusisto, A.; and Lutz, C. 2024. Logical characterizations of recurrent graph neural networks with reals and floats. In Globersons, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J. M.; and Zhang, C., eds., *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Barceló, P.; Kozachinskiy, A.; Lin, A. W.; and Podolskii, V. V. 2024. Logical languages accepted by transformer encoders with hard attention. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Barceló, P.; Kostylev, E. V.; Monet, M.; Pérez, J.; Reutter, J.; and Silva, J. P. 2020. The logical expressiveness of graph neural networks. In *International Conference on Learning Representations*.

Benedikt, M.; Lu, C.; Motik, B.; and Tan, T. 2024. Decidability of graph neural networks via logical characterizations. In *51st International Colloquium on Automata, Languages, and Programming, ICALP*.

Bongini, P.; Bianchini, M.; and Scarselli, F. 2021. Molecular generative graph neural networks for drug discovery. *Neurocomputing* 450:242–252.

Chiang, D.; Cholak, P.; and Pillay, A. 2023. Tighter bounds on the expressivity of transformer encoders. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, 5544–5562. PMLR.

Cucala, D. T.; Grau, B. C.; Motik, B.; and Kostylev, E. V. 2023. On the correspondence between monotonic max-sum gnns and datalog. In Marquis, P.; Son, T. C.; and Kern-Isberner, G., eds., *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2023, Rhodes, Greece, September 2-8, 2023*, 658–667.

Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; and Dahl, G. E. 2017. Neural message passing for quantum chemistry. In Precup, D., and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, 1263–1272. PMLR.

Grohe, M. 2021. The logic of graph neural networks. In *36th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2021, Rome, Italy, June 29 - July 2, 2021*, 1–17. IEEE.

Hague, M.; Lin, A. W.; and Hong, C. 2019. CSS minification via constraint solving. *ACM Trans. Program. Lang. Syst.* 41(2):12:1–12:76.

Jaakkola, R.; Janhunen, T.; Kuusisto, A.; Ortiz, M.; Selin, M.; and Simkus, M. 2025. Graph learning via logic-based weisfeiler-leman variants and tabularization. *CoRR* abs/2508.10651.

Kipf, T.; Fetaya, E.; Wang, K.-C.; Welling, M.; and Zemel, R. 2018. Neural relational inference for interacting systems. In Dy, J., and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 2688–2697. PMLR.

Kriege, N. M.; Johansson, F. D.; and Morris, C. 2020. A survey on graph kernels. *Appl. Netw. Sci.* 5(1):6.

Nunn, P.; Sälzer, M.; Schwarzentruber, F.; and Troquard, N. 2024. A logic for reasoning about aggregate-combine graph neural networks. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, 3532–3540. ijcai.org.

Rosenbluth, E.; Tönshoff, J.; and Grohe, M. 2023. Some might say all you need is sum. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, 4172–4179. ijcai.org.

Schönherr, M., and Lutz, C. 2025. Logical characterizations of gnns with mean aggregation. *CoRR* abs/2507.18145.

Seidl, H.; Schwentick, T.; and Muscholl, A. 2008. Counting in trees. In Flum, J.; Grädel, E.; and Wilke, T., eds., *Logic and Automata: History and Perspectives [in Honor of Wolfgang Thomas]*, Texts in Logic and Games, 575–612. Amsterdam University Press.

Shervashidze, N.; Schweitzer, P.; van Leeuwen, E. J.; Mehlhorn, K.; and Borgwardt, K. M. 2011. Weisfeiler-lehman graph kernels. *J. Mach. Learn. Res.* 12:2539–2561.

Shlomi, J.; Battaglia, P.; and Vlimant, J.-R. 2020. Graph neural networks in particle physics. *Machine Learning: Science and Technology* 2(2):021001.

Sälzer, M.; Köcher, C.; Kozachinskiy, A.; Zetzsche, G.; and Lin, A. W. 2026. The counting power of transformers. In *The Fourteenth International Conference on Learning Representations*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph attention networks.

Wu, S.; Sun, F.; Zhang, W.; Xie, X.; and Cui, B. 2022. Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.* 55(5).

Yang, A.; Chiang, D.; and Angluin, D. 2024. Masked hard-attention transformers recognize exactly the star-free languages. In Globersons, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J. M.; and Zhang, C., eds., *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Zhang, M.; Li, P.; Xia, Y.; Wang, K.; and Jin, L. 2021. Labeling trick: A theory of using graph neural networks for multi-node representation learning. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y. N.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, 9061–9073.

Zhang, J.; Liu, C.; and Jiang, X. 2024. Polynomial kernel learning for interpolation kernel machines with application to graph classification. *Pattern Recognit. Lett.* 186:7–13.

Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2020. Graph neural networks: A review of methods and applications. *AI Open* 1:57–81.