debanjandutta_r
Debanjan Dutta
JRF(CS), ECSU, ISI

On Logical Characterisation of Graph Neural Network

Instructor: Dr Sujata Ghosh
Deadline: Apr 17, 2024
Sub on: Apr 16, 2024 9:26pm

Abstract

This project investigates the expressive power of Graph Neural Networks (GNNs) by establishing connections between their reasoning capabilities and different logic classes. The analysis of the representational power of GNNs through the lens of logic relies on the classification (`true` or `false`) of a node-colour in a graph or a suitable logical formula evaluating to a property based on the color of a node. Ultimately, the project seeks to selectively characterise the structure of the GNNs in such a way that modifications to the components of its structure results into establishing the equivalence (or, proper inclusion) between different classes of logic formalism.

1 Introduction

Graph Neural Network (GNN) has gained a tremendous popularity in various real world task that typically comes under categories such as node classification, edge prediction and graph morphisms. A Weisfeiler Leman (WL) test performs partitioning nodes in a graph depending on the heuristic it follows. The partition is done under the well-known paradigm of graph coloring. The key observation that a WL-Test provides is – after partitioning the vertex set of two graphs independently, if the test outputs do not agree on the partitioning the given graphs are non-isomorphic. On such a backdrop, there exists a rigours study connecting a fragment of first order logic, often termed as FOC_2 class with the execution of WL test. Thus,[1] a question that is natural to ask, is how well the GNN structures are associated to FOC_2.

The rudimentary structure of a graph neural network depends on the functions called aggregation and combination. Such an AC-GNN takes an one-hot embedding of nodes as parameter and executes classification after a predefined number of iterations over all the embedded nodes. As the structure gets defined in section 2, it can be shown that the AC-GNN is strictly less powerful than FOC_2 logic class, while being equivalent to the expressiveness of graded modal logic – a fragment of FOC_2 logic that is constrained to be *local* as defined in section 2.

In contrary, if a *global* $\mathrm{ReadOut}$ function is applied to nodes in every iteration (calling such a model as ACR-GNN) or even only in final iteration (calling such a model as AC-FR-GNN), the expressiveness of such models can capture FOC_2 logic. However, the other side of the inclusion has not been studied. A *local* (or, *global*) characteristic of a function or logical expression applied to a node in a graph hint to whether that entity is capturing information only from the neighbourhood of that node (or, considering all the nodes).

This project is built on a recent exploration by Barceló et al. in *The Logical Expressiveness of Graph Neural Networks*. We will be covering only a portion that is related to AC-GNN only.

2 The Model and its Variants

This section deals with defining the model and it's properties.

2.1 Aggregate-Combine Graph Neural Network

A graph neural network (GNN) is a dynamics map $\mathcal{F} : \mathbb{Q}^{n\times d} \to \mathbb{Q}^{n\times d}$ where n is the number of vertices in the graph G and d be the embedding dimension of the vertex. Given a finite numbers of colours (d), each vertex v is embedded as a eye vector $\mathbb{1}_{\mathrm{Col}(v)}$ where all the components are 0 except at the index $\mathrm{Col}(v)$ i.e the color of vertex v. Having said that we are good to proceed with the definition of a GNN as a node-color classifier.

Definition 2.1 (Model - Aggregate Combine GNN)**.** *Given a graph G, a vertex v and $L \in \mathbb{N}$; the **A**ggregate **C**ombine-**G**raph **N**eural **N**etwork (AC-GNN) model $\mathcal{A}$ performs a partitioning task (similar to vertex colouring) on v as follows:*

$$\boldsymbol{x}_v^{(l)} = \mathrm{COM}^{(l)}\left(\boldsymbol{x}_v^{(l-1)}, \mathrm{AGG}^{(l)}\left(\{\!\{\boldsymbol{x}_u^{(l-1)} \mid \boldsymbol{x}_u \in \mathcal{N}(v)\}\!\}\right)\right) \quad l \in [L] \textit{ and } \boldsymbol{x}_v^{(0)} = \mathbb{1}_{\mathrm{Col}(v)} \tag{1}$$

[1]Even though there are established results on the equivalence of performance between WL-Test and GNNs, we can look into the practical applications of GNNs which may lead to such questions.

where $\mathrm{AGG}^{(l)}$ *is an aggregation function (in this case, it a point-wise sum) applied on the embedding of the vertices neighbour to* v *at some iteration* $l \in [L]$ *and* $\mathrm{COM}^{(l)}$ *is a combination function which may vary from being a learnable function, weighted sum but in this case it is restricted to some*

$$\mathrm{COM}^{(l)}(\boldsymbol{v}, \boldsymbol{u}) = f(\boldsymbol{A}^{(l)}\boldsymbol{v} + \boldsymbol{C}^{(l)}\boldsymbol{u} + \boldsymbol{b})$$

where f *can be any non-linearity which is* $\min(\max(x, 0), 1)$ *in this case. Let* $\boldsymbol{X}_G^{l)}$ *be the set of all vertex-embeddings in the graph at* l*-th iteration. Then*

$$\mathcal{F}^l = \mathcal{F}(\boldsymbol{X}_G^{l)}) = \{\!\{ \boldsymbol{x}_v^{(l+1)} \mid \textit{ for all vertex } v \textit{ in } G \}\!\}.$$

Finally, given a function CLS *the node classification of a vertex* v *in* G *is defined by*

$$\mathcal{A}(G, v) = \mathrm{CLS}(\boldsymbol{x}_v^{(L)}) \in \{\texttt{true}, \texttt{false}\}.$$

Before defining another model of our use, let us characterise the property of AC-GNN. An iteration in a AC-GNN can only capture the *local* behaviour of nodes being processed meaning throughout the predefined number of iterations if an information does not flow through the neighborhood of a vertex, there will be a loss of information that to procure true bature of that vertex – this is exactly what described in Proposition 3.1. To add to this, we will be calling a model homogeneous if all constituent functions AGG and COM are kept identical throughout all the iterations.

Now we are ready to define another model – Aggregate Combine Readout GNN as a node-colour classifier, overcoming the short-coming of the earlier one.

Definition 2.2 (Model - Aggregate Combine Readout GNN)**.** *A node-colour classifier AC-GNN* $\mathcal{A}$ *is called a ACR-GNN if the update function Eqn 1 is defined as follows*

$$\boldsymbol{x}_v^{(l)} = \mathrm{COM}^{(l)}\left(\boldsymbol{x}_v^{(l-1)}, \mathrm{AGG}^{(l)}\left(\{\!\{\boldsymbol{x}_u^{(l-1)} \mid \boldsymbol{x}_u \in \mathcal{N}(v)\}\!\}\right), \mathrm{ReadOut}^{(l)}\left(\{\!\{\boldsymbol{x}_u^{(l-1)} \mid \boldsymbol{x}_u \in G\}\!\}\right)\right) \tag{2}$$

where, $l \in [L]$ *and* $\boldsymbol{x}_v^{(0)} = \mathbb{1}_{\mathrm{Col}(v)}$ *and the combination function is modified as*

$$\mathrm{COM}^{(l)}(\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{w}) = f(\boldsymbol{A}^{(l)}\boldsymbol{x} + \boldsymbol{C}^{(l)}\boldsymbol{u} + \boldsymbol{R}^{(l)}\boldsymbol{w} + \boldsymbol{b}).$$

Keeping the homogeneity property identical as earlier, now let us see the following execution on vertex v_1 for a path graph $K_{3,3}$:

Now we can define the relationship between a node-colour classification model with a logical formula:

Definition 2.3. *A node-colour classifier GNN* $\mathcal{A}$ *captures a logical formula* $\varphi(x)$ *if for every graph* G *and node* v *in* G*,* $\mathcal{A}(G, v) = \texttt{true}$ *if and only if* $(G, v) \models \varphi$*.*

2.2 Some selected families of logic

In this subsection, we will be discussing the class of logic and its properties that are required to express the representation capability of the above mentioned models. Instead of defining the subclasses of FO logic formally, let us have an example as follows:

$$\alpha(x) = \mathrm{Green}(x) \wedge \exists y(\neg E(x, y) \wedge \mathrm{Blue}(y)) \wedge \exists z(E(x, z) \wedge \mathrm{Red}(z)).$$

The logical formula α involves three variables. We can introduce a subclass of FO formulas, FO_2 that is expressible by two variables only. For example, we can express α in FO_2 by replacing z with y. Unfortunately, we cannot have an equivalent formula to α having less number of variables.

Now let us have another example of first order formula

$$\beta(x) = \mathrm{Green}(x) \wedge \exists y(\neg E(x, y) \wedge \mathrm{Blue}(y)) \wedge \exists z_1 \exists z_2(E(x, z_1) \wedge E(x, z_2) \wedge \neg(z_1 = z_2) \wedge \mathrm{Red}(z_1) \wedge \mathrm{Red}(z_2)).$$

This formula ensures existence of three distinct vertices y, z_1, z_2 where y is a non-neighbour to x and z_1, z_2 are neighbour to x which is coloured as red. In this formula we can apply the earlier trick to get a β involving 3 variables but cannot have no less than that. To this end, let us consider an operator $\exists^{\geq n}$ implying existence of at least $n \geq 1$ *distinct* variable satisfying the succeeding formula. The segment of FO_2 logic that involves such a *counting quantifier* is known as FOC_2 logic class. As an example we can consider the following

$$\gamma(x) = \mathrm{Green}(x) \wedge \exists^{\geq 2} y \, \mathrm{Blue}(y).$$

Now if every bound variable in a FOC_2 formula involves an edge constraints with it we will call them *graded* modal logic. Thus a graded modal formula φ is of form:

$$\varphi(x) : \text{Col}(x) \mid \neg\varphi(x) \mid \varphi(x) \wedge \psi(x) \mid \exists^{\geq n} y(E(x,y) \wedge \varphi(x)). \tag{3}$$

CLearly, the logical expressiveness of FOC_2 is strictly less than that of FO_2 and similarly, graded modal logic shares a portion of FOC_2 logic class. As we have introduced, we will be noticing the equivalence of the node-colour classifier model with such family of logics.

3 Logical Expressiveness of Node-Colour Classifier AC-GNN

This section deals with formally characterising the expressive power of AC-GNN only. For models like ACR-GNN it involves discussions that may not befit the stipulated time frame.

Proposition 3.1. *There is an FOC_2 classifier that is not captured by any AC-GNN.*

Proof. Let us consider an FO_2 formula $\delta(v) = \text{Red}(v) \wedge \exists u\, \text{Blue}(u)$. To prove the above statement using contradiction let us consider, an AC-GNN $\mathcal{A}$ with number of layers $L \in \mathbb{N}$ can satisfy $\delta(v)$ that is for any graph G, $\mathcal{A}(G, v) \models \delta$. Then, we consider a path graph P_{L+2} of length $L+2$ such that to vertices $v, u_1, \ldots, u_L$ are all red except for the vertex u_{L+1} which is blue. No matter what choice of AGG, COM and CLS be, $\mathcal{A}(P_{L+2}, v) = \texttt{false}$ although δ is satisfiable in such a P_{L+2}. Thus a contradiction.

Suppose, for adding more power to AC-GNN we let the GNN run for at least $|G.E|$ times or $f(|G.E|)$ times . Then the above statement holds. The trick[2] is to construct a disconnected graph having all red nodes in one component (where the vertex v lies) and a single blue node in another. For such a graph, although $\mathcal{A}(G, v) = \texttt{false}$, $(G, v) \models \delta(v)$.

Now, it not hard to see why introducing counting quantifiers would not also help in terms of capturing the logical representational power of AC-GNN. □

Thus the apparent questions are - (1) what are the family of logic that are representaed by an AC-GNN and (2) what class of node-colour classification models can capture the FOC_2 logic family? We will answer the first question in Theorem 3.4. To answer the second question it is claimed that ACR-GNN can express FOC_2 class of logic – however, the proof is out of scope of this project.

Proposition 3.2. *Each graded modal logic classifier is captured by a simple homogeneous AC-GNN.*

Proof. Please recall the syntax of graded modal logic as given in Eqn 3. Because, we are trying to construct an AC-GNN $\mathcal{A}_\varphi$ satisfying φ we are free to choose the embedding dimension of the nodes. Suppose the graded modal logic formula φ involves L sub-formulas $(\varphi_1, \varphi_2, \ldots, \varphi_L)$ such that if φ_k is a sub-formula of φ_l then $k < l$. We will be considering the embedding dimension of each node to be L that is the number of sub-formula present in it. Also, it is easy to apprehend that the number of iterations the model $\mathcal{A}_\varphi$ need should not be more than L. The ℓ^{th} component of the feature vector of node v at iteration ℓ, $(\boldsymbol{x}_v^{(\ell)})_\ell$ should evaluate to 1 if and only if the formula φ_ℓ is satisfied at node v. Note that $\varphi_L = \varphi$ and hence $(\boldsymbol{x}_v^{(L)})_L = 1$ for all the nodes v if and only if φ is satisfied at the node.

Now the construction of the homogeneous functions viz.

$$\text{AGG}(\boldsymbol{U}) = \sum_{\boldsymbol{u} \in \boldsymbol{U}} \boldsymbol{u}$$

$$\text{COM}(\boldsymbol{v}, \boldsymbol{u}) = f\left(\boldsymbol{A}\boldsymbol{v} + \boldsymbol{C}\boldsymbol{u} + \boldsymbol{b}\right) \text{ where } \boldsymbol{A}, \boldsymbol{C} \in \mathbb{R}^{L \times L} \text{ and } \boldsymbol{b} \in \mathbb{R}^L \text{ and } f(x) = \min(\max(x, 0), 1)$$

is all that we need show to ensure existence of $\mathcal{A}_\varphi$. Let us then fill up the matrices $\boldsymbol{A}$ and $\boldsymbol{C}$ and bias $\boldsymbol{b}$ as follows:

$Case\ 0.$ if $\varphi_\ell(v) = \text{Col}(v)$ with Col one of the (base) colors, then $A_{\ell\ell} = 1$,
$Case\ 1.$ if $\varphi_\ell(v) = \varphi_j(v) \wedge \varphi_k(v)$ then $A_{j\ell} = A_{k\ell} = 1$ and $\boldsymbol{b}_\ell = -1$,
$Case\ 2.$ if $\varphi_\ell(v) = \neg\varphi_k(v)$ then $A_{k\ell} = -1$ and $\boldsymbol{b}_\ell = 1$,
$Case\ 3.$ if $\varphi_\ell(v) = \exists^{\geq n}(E(v,u) \wedge \varphi_k(u))$ then $C_{k\ell} = 1$ and $\boldsymbol{b}_\ell = -n + 1$

Now unrolling the update equation 1 as

$$\boldsymbol{x}_v^{(i)} = f\left(\boldsymbol{A}\boldsymbol{x}_v^{(i-1)} + \sum_{u \in \mathcal{N}(v)} \boldsymbol{C}\boldsymbol{x}_u^{(i-1)} + \boldsymbol{b}\right)$$

the formulas can be satisfied directly. □

[2] IThe paper suggests a different one which I could not understood.

Proposition 3.3. *The logical expression an AC-GNN can express is exactly what captured by a graded modal logic.*

Proof. We would take help of the following propositions:

1. If the WL test assigns the same color to two nodes in a graph, then every AC-GNN classifies either both nodes as true or both nodes as `false`. [Xu et al., 2018]

2. Let α be a unary FO formula. If α is not equivalent to a graded modal logic formula then there exist two graphs G, G' and two nodes v in G and u' in G' such that $\text{Unrv}_G^L(v) \simeq$ isomorphic $\text{Unrv}_{G'}^L(u')$ for every $L \in \mathbb{N}$ and such that $u \models \alpha$ in G but $u' \not\models \alpha$ in G'. [Otto, 2019].

Let G be a graph (simple, undirected and node-colored), v be a node in G, and $L \in \mathbb{N}$. The unravelling of v in G at depth L, denoted by $\text{Unrv}_G^L(v)$, is the (simple undirected nodecolored) graph that is the tree having:

- a node $(v, u_1, \ldots, u_i)$ for each path $(v, u_1, \ldots, u_i)$ in G with $i \leq L$,
- an edge between $(v, u_1, \ldots, u_{i-1})$ and $(v, u_1, \ldots, u_i)$ when $\{u_{i-1}, u_i\}$ is an edge in G (assuming that u_0 is v), and
- each node $(v, u_1, \ldots, u_i)$ colored the same as u_i in G.

Then we can observe that

3. Let G and G' be two graphs, and v and v' be two nodes in G and G', respectively. Then for every $L \in \mathbb{N}$, the WL test assigns the same color to v and v' at round L if and only if there is an isomorphism between $\text{Unrv}_G^L(v)$ and $\text{Unrv}_{G'}^L(v')$ sending v to v'.

Now from observation 1 and 3, we can conclude the follow:

4. Let G and G' be two graphs with nodes v in G and v' in G' such that $\text{Unrv}_G^L(v)$ is isomorphic to $\text{Unrv}_{G'}^L(v')$ for every $L \in \mathbb{N}$. Then for any AC-GNN $\mathcal{A}$, we have $\mathcal{A}(G, u) = \mathcal{A}(G', u')$.

We will now prove the contrapositive of the statement to be proved – If a logical classifier α is not equivalent to any graded modal logic formula, then there is no AC-GNN that captures α.

Let α be a logical classifier that is not equivalent to any graded modal logic formula. To contradict, let us assume there exists an AC-GNN $\mathcal{A}_\alpha$ capturing α. By observation 2, there exist two graphs G, G' and two nodes v in G and u' in G' such that i) $\text{Unrv}_G^L(v) \simeq \text{Unrv}_{G'}^L(u')$ for every $L \in \mathbb{N}$ and ii) such that $u \models \alpha$ in G but $u' \not\models \alpha$ in G'. But as i) holds by observation 4, we can say $\mathcal{A}(G, u) = \mathcal{A}(G', u')$ – violating to ii) and hence the contradiction that $\mathcal{A}_\alpha$ can capture α. ☐

Theorem 3.4. *A logical classifier is captured by AC-GNNs if and only if it can be expressed in graded modal logic.*

Proof. Using Propositions 3.2 and 3.3, we can conclude to this. ☐

4 Conclusion

This project tried to explore the established result by Barceló et al. that bridges the gap between the theoretical representational power of a graph neural network and its wide application in tasks coming under the category of node classification. The distinction between local and global information processing is the key idea that segregates the models in terms of their logical expressiveness. One of the obvious extensions could be introducing FOC_k logic class and study the expressiveness of the proposed models. Seeming the study of logical expressiveness of AC-GNN depends on the well-defined studies of WL-test, graded modal logic and connection with WL-test and AC-GNN. However, had we extend graded modal logic by allowing a path of length 2 instead of 1 that is an edge and allowing AC-GNN to access information at depth 2, instead of aggregating the neighbourhood information, can we conclude similar result – this may worth investigation for further studies.

References

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

Martin Otto. Graded modal logic and counting bisimulation. *ArXiv*, abs/1910.00039, 2019. URL https://api.semanticscholar.org/CorpusID:203610381.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *ArXiv*, abs/1810.00826, 2018. URL https://api.semanticscholar.org/CorpusID:52895589.