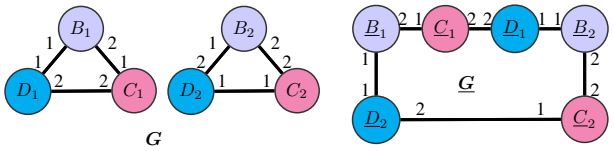
A. Supplementary material

We now provide detailed proofs for all our propositions and lemmas.

Proof of Proposition 1

Proof. We show that CPNGNN, using some consistent port ordering, can distinguish some non-isomorphic graphs that LU-GNNs cannot.

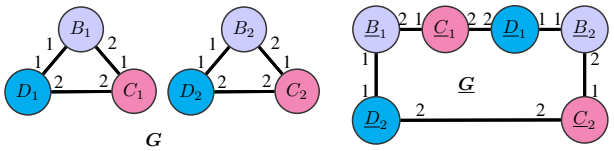


We construct a pair of graphs $\boldsymbol{G}$ and $\underline{\boldsymbol{G}}$ such that $\boldsymbol{G}$ consists of two triangles that differ in port-ordering but are otherwise identical, while $\underline{\boldsymbol{G}}$ (indicated by underlined symbols) consists of a single even-length cycle. The construction ensures that each node labeled with $\boldsymbol{X} \in \{\boldsymbol{B_1}, \boldsymbol{C_1}, \boldsymbol{D_1}, \boldsymbol{B_2}, \boldsymbol{C_2}\}$ in $\boldsymbol{G}$ has the same identical view (i.e., indistinguishable node features, and neighborhood) as the corresponding node labeled $\underline{\boldsymbol{X}}$ in $\underline{\boldsymbol{G}}$. However, $\boldsymbol{D_2}$ and $\underline{\boldsymbol{D_2}}$ have distinguishable neighborhoods due to different port-numbers: e.g., $\boldsymbol{D_2}$ is connected to $\boldsymbol{B_2}$ at port 2, whereas $\underline{\boldsymbol{D_2}}$ is connected to $\underline{\boldsymbol{B_1}}$ at port 1. Likewise, $\boldsymbol{D_2}$ is connected to $\boldsymbol{C_2}$ at port 1, in contrast to $\underline{\boldsymbol{D_2}}$ that is connected to $\underline{\boldsymbol{C_2}}$ at port 2. However, LU-GNN does not incorporate any spatial information such as ports, and fails to tell one graph from the other. □

Note that since $\angle B_1 C_1 D_1$ differs from $\angle \underline{B}_1 \underline{C}_1 \underline{D}_1$, DimeNet can also distinguish between the two graphs.

Proof of Proposition 2

Proof. We now illustrate the importance of choosing a good consistent port numbering. Specifically, we construct a pair of graphs, and two different consistent port numberings p and q such that CPNGNN can distinguish the graphs with p but not q.



We modify the consistent port numbering from the construction of Proposition 1. We consider the same pair of graphs as in the proof of Proposition 1. However, instead of having different numberings for the two components (i.e., triangles) of $\boldsymbol{G}$, we now carry over the ordering from one component to the other. The two components become identical with this modification. For any node labeled $\underline{X}_1$ or $\underline{X}_2$, and any neighbor labeled $\underline{Y}_1$ or $\underline{Y}_2$, $X, Y \in \{B, C, D\}$, we can now simply assign the same respective local ports as the nodes labeled X_1 and Y_1 (or, equivalently, X_2 and Y_2). It is easy to verify that the two graphs become port-locally isomorphic under the new ordering, and thus cannot be separated with any permutation-invariant readout (using Proposition 3).

□

Proof of Proposition 3

Proof. We begin with the following definition.

Definition 3. Two nodes in a graph are *locally indistinguishable* if they have identical feature vectors and identical port-ordered neighborhoods.

In other words, for locally indistinguishable nodes u and v, not only are their neighbors identical but also the respective ports that connect u and v to their identical neighbors are identical.

If the surjection $f : V_1 \to V_2$ in Definition 2 is also injective, then we can simply take $h = f$. Therefore, we focus on the case when f is not injective. We will show that f can be used to inform h. Since f is not injective, there exist $v_1, v_1' \in V_1$ such that $v_1 \neq v_1'$ but $f(v_1) = f(v_1') = v_2$ for some $v_2 \in V_2$. Then, by condition (a) in Definition 2, we immediately get that the feature vector

$$x_{v_1} = x_{f(v_1)} = x_{f(v_1')} = x_{v_1'} \,. \tag{6}$$

Moreover, by other conditions, there is a consistent port bijection from neighborhood of v_1 to that of v_2, and likewise another bijection from neighborhood of v_1' to that of v_2. Therefore, there is a consistent port bijection from neighborhood of v_1 to that of v_1'. Together with (6) and our assumption that $f(v_1) = f(v_1') = v_2$, this implies that v_1 and v_1' are locally indistinguishable. Note that there could be more such nodes that are indistinguishable from v_1 (or v_1'), e.g., when all such nodes map to v_2 as well.

Without loss of generality, let $\mathcal{E}_1(v_1) \subseteq V_1$ denote the equivalence class of all nodes, including v_1, that are indistinguishable from v_1 in graph G_1. Similarly, let $\mathcal{E}_2(v_2) \subseteq V_2$ be the class of nodes indistinguishable from v_2 in G_2. Consider $\ell_1 = |\mathcal{E}_1(v_1)|$ and $\ell_2 = |\mathcal{E}_2(v_2)|$. We claim that $\ell_1 = \ell_2$. Suppose not. Then if $\ell_1 < \ell_2$, we can have h map each node in $\mathcal{E}_1(v_1)$ to a separate node in $\mathcal{E}_2(v_2)$, and use the same mapping as f on the other nodes in V_1. Doing so does not decrease the co-domain of

V_2, and h remains surjective. We are therefore left with $\ell_2 - \ell_1 > 0$ nodes from $\mathcal{E}_2(v_2)$. Therefore, these nodes must have at least one preimage in the set $V_1 - \mathcal{E}_1(v_1)$ since f (and thus h) is a surjection by assumption (**a**) in Definition 1. This is clearly a contradiction since any such preimage must have either a different feature vector, or a non-isomorphic port-consistent neighborhood. By a symmetric argument, using the surjection of map from V_2 to V_1, we conclude that $\ell_1 = \ell_2$. Note that h did not tinker with the nodes that were outside the class $\mathcal{E}_1(v_1)$. Recycling the procedure for other nodes in $V_1 - \mathcal{E}_1(v_1)$ that might map under f to a common image in V_2, we note that h ends up being injective. Since h remains surjective throughout the procedure, we conclude that h is a bijection.

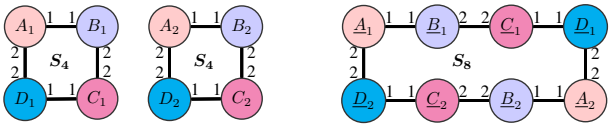
We now prove by induction that the corresponding nodes in port-locally isomorphic graphs have identical embeddings for any CPNGNN. Consider any such GNN with $L+1$ layers parameterized by the sequence $\theta_{1:L+1} \triangleq (\theta_1, \ldots, \theta_L, \theta_{L+1})$. Since there exists a bijection h such that any node $v_1 \in G_1$ has an identical local view (i.e., node features, and port-numbered neighbors) as $v_2 = h(v_1) \in G_1$, the updated embeddings for v_1 and v_2 are identical after the first layer. Assume that these embeddings remain identical after update from each layer $\ell \in \{2, 3, \ldots, L\}$. Since v_1 and v_2 have identical local views and have identical embedding from the Lth layer, the updates for these nodes by the $(L+1)th$ layer are identical. Therefore, v_1 and v_2 have identical embeddings. Since h is a bijection, for every $v \in V_1$ there is a corresponding $h(v) \in V_2$ with the same embedding, and thus both G_1 and G_2 produce the same output with any permutation readout function. Our choice of $\theta_{1:L+1}$ was arbitrary, so the result follows.

□

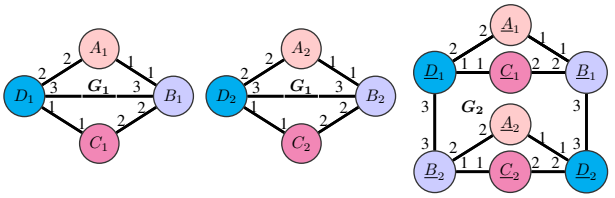
Proof of Proposition 4

Proof. We now show that there exist consistent port orderings such that CPNGNNs with permutation-invariant readout cannot decide several important graph properties: girth, circumference, diameter, radius, conjoint cycle, total number of cycles, and k-clique. The same result also holds for LU-GNNs where nodes do not have access to any consistent port numbering.

We first construct a pair of graphs that have cycles of different length but produce the same output embedding via the readout function. Specifically, we show that CPNGNNs cannot decide a graph having cycles of length n from a cycle of length $2n$. We construct a counterexample for $n = 4$. Our first graph consists of two cycles of length 4 (each denoted by S_4), while the other graph is a cycle of length 8 (denoted by S_8). We associate identical feature vectors with nodes that have the same color, or equivalently, that are marked with the same symbol ignoring the subscripts and the underline. For example, A_1, A_2, $\underline{A}_1$, and $\underline{A}_2$ are all assigned the same feature vector. Moreover, we assign identical edge feature vectors to edges that have the same pair of symbols at the nodes.



Thus, we note that a bijection exists between the two graphs with node X in the first graph corresponding to $\underline{X}$ in the second graph such that both the nodes have identical features and indistinguishable port-ordered neighborhoods. Since, the two graphs have different girth, circumference, diameter, radius, and total number of cycles, it follows from Proposition 3 that CPNGNN cannot decide these properties. Note that the graph with two S_4 cycles is disconnected, and hence its radius (and diameter) is ∞.



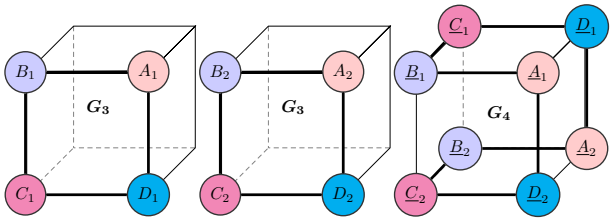
We craft a separate construction for the remaining properties, namely, k-clique and conjoint cycle. The main idea is to replicate the effect of the common edge in the conjoint cycle via two identical components of another graph (that does not have any conjoint cycle) such that the components are cleverly aligned to reproduce the local port-ordered neighborhoods and thus present the same view to each node (see the adjoining figure). Specifically, each conjoint cycle is denoted by G_1, and the other graph that does not have any conjoint cycles by G_2. The graphs, being port-locally isomorphic, are indistinguishable by CPNGNN.

For the k-clique, we simply connect A_1 to C_1, A_2 to C_2, $\underline{A}_1$ to $\underline{C}_1$, and $\underline{A}_2$ to $\underline{C}_2$ via a new port 3 at each of these nodes. Doing so ensures that the new graphs are port-locally isomorphic as well. Adding these edges, we note that, unlike G_2, each conjoint cycle G_1 yields a 4-clique. □

Proof of Proposition 5

Proof. We now demonstrate the representational limits of DimeNets. Specifically, we show two graphs that differ in several graph properties such as girth, circumference, diameter, radius, or total number of cycles. However, these graphs cannot be distinguished by DimeNets.

Note that DimeNet will be able to discriminate S_8 from the graph with cycles S_4 (recall our construction in Proposition 4), since, e.g., $\angle B_1C_1D_1$ in S_4 is different from $\angle \underline{B}_1\underline{C}_1\underline{D}_1$ in S_8. In order to design a failure case for DimeNet, we need to construct a pair of non-isomorphic graphs that have not only identical local pairwise distances but also angles, so that their output embedding is same.



Our idea is to overlay the cycles S_4 and S_8 on a cube (see G_3 and G_4 - the graphs consist of only edges in bold). Doing so does not have any bearing on the graph properties. Since we orient the edges of these cycles along the sides of the cube, the local distances are identical. Moreover, by having $A_1B_1C_1D_1$ and $A_2B_2C_2D_2$ as opposite faces of the cube, we ensure that each angle in G_4 is a right angle, exactly as in G_3. Thus, for each $X \in \{A, B, C, D\}$, nodes X_1, X_2, $\underline{X}_1$, and $\underline{X}_2$ have identical feature vectors and identical local spatial information. Thus, the embeddings for X_1, X_2, $\underline{X}_1$, and $\underline{X}_2$ are identical, and any permutation-invariant readout results in identical output embeddings for the two graphs.

□

Proof of Proposition 6

Proof. We now show that the complexity of the GNN may be bounded by the complexity of the computation trees. In other words, the worst case generalization bound over a set of graphs corresponds to having each graph be a single computation tree. Formally,

$$\begin{aligned}
\hat{\mathcal{R}}_{\mathcal{G}} &\triangleq \mathbb{E}_\sigma \sup_\Theta \sum_{j=1}^{m} \sigma_j f(G_j;\Theta) \\
&= \mathbb{E}_\sigma \sup_\Theta \sum_{j=1}^{m} \sigma_j \mathbb{E}_{T\sim w'(G_j)} f_c(T;\Theta) \\
&\leq \mathbb{E}_\sigma \mathbb{E}_{t_1,..t_m} \sup_\Theta \sum_{j=1}^{m} \sigma_j f_c(t_j;\Theta) \\
&= \mathbb{E}_{t_1,..t_m} \underbrace{\mathbb{E}_\sigma \sup_\Theta \sum_{j=1}^{m} \sigma_j f_c(t_j;\Theta)}_{\hat{\mathcal{R}}_{\mathcal{T}}} ,
\end{aligned}$$

where we invoked Jensen's inequality to swap the expectation with supremum for our inequality (the operation is permissible since sup is a convex function). □

Proof of Lemma 2

Proof. Our objective here is to bound the effect of change in weights from (W_1, W_2) to (W_1', W_2') on the embedding of the root node of our fixed tree (that has depth L). Since non-linear activation and permutation-invariant aggregation are both Lipschitz-continuous functions, and the feature vector at the root x_L and the weights have bounded norm, the embedding at the root of the tree adapts to the embeddings from the subtrees.

Specifically, we note that the l_2-norm of difference of embedding vectors produced by (W_1, W_2) and (W_1', W_2') is

$$\begin{aligned}
\Delta_L &\triangleq ||\,T_L(W_1,W_2) - T_L(W_1',W_2')||_2 \\
&= \Big|\Big|\phi\big(W_1 x_L + W_2\, \rho\big(\underbrace{\sum_{j\in C(x_L)} g(T_{L-1,j}(W_1,W_2))}_{\triangleq\, R(W_1,W_2,x_L)}\big)\big) \\
&- \phi\big(W_1' x_L + W_2' \rho\big(\sum_{j\in C(x_L)} g(T_{L-1,j}(W_1',W_2'))\big)\big)\Big|\Big|_2 \\
&\leq C_\phi \,\big|\big|(W_1 - W_1')x_L\big|\big|_2 \qquad (7) \\
&+ C_\phi ||W_2 R(W_1,W_2,x_L) - W_2' R(W_1',W_2',x_L)||_2 \,.
\end{aligned}$$

Therefore, in order to find an upper bound for Δ_L, we will bound the two terms in the last inequality separately. We first bound the second term using the sum of $||W_2 R(W_1,W_2,x_L) - W_2' R(W_1,W_2,x_L)||_2$ and $||W_2' R(W_1,W_2,x_L) - W_2' R(W_1',W_2',x_L)||_2$. Note that

$$\begin{aligned}
&||W_2' R(W_1,W_2,x_L) - W_2' R(W_1',W_2',x_L)||_2 \\
\leq\; &||W_2'||_2 \;\; ||R(W_1,W_2,x_L) - R(W_1',W_2',x_L)||_2 \,.
\end{aligned} \qquad (8)$$

Since g is C_g-Lipschitz, the branching factor of tree is d, and ρ is C_ρ-Lipschitz, therefore, R is dC_gC_ρ-Lipschitz. We will use this fact to bound (8). Specifically,

$$\begin{aligned}
&||R(W_1,W_2,x_L) - R(W_1',W_2',x_L)||_2 \\
&\leq C_\rho \Big|\Big| \sum_{j\in C(x_L)} \Big(g(T_{L-1,j}(W_1,W_2)) \\
&\qquad\qquad - g(T_{L-1,j}(W_1',W_2'))\Big)\Big|\Big|_2 \\
&\leq C_\rho \sum_{j\in C(x_L)} \Big|\Big|\Big(g(T_{L-1,j}(W_1,W_2)) \\
&\qquad\qquad - g(T_{L-1,j}(W_1',W_2'))\Big)\Big|\Big|_2 \\
&\leq C_\rho C_g \sum_{j\in C(x_L)} \Big|\Big| T_{L-1,j}(W_1,W_2) - T_{L-1,j}(W_1',W_2')\Big|\Big|_2 \\
&= C_\rho C_g \sum_{j\in C(x_L)} \Delta_{L-1,j} \,.
\end{aligned}$$

Using this with $||W_2'||_2 \leq B_2$ in (8), we immediately get

$$\begin{aligned} ||W_2'R(W_1,W_2,x_L) &- W_2'R(W_1',W_2',x_L)||_2 \\ &\leq B_2C_\rho C_g \sum_{j\in C(x_L)} \Delta_{L-1,j} \\ &\leq B_2C_\rho C_g d \max_{j\in C(x_L)} \Delta_{L-1,j}\,. \end{aligned}$$

In other words, we bound the effect on each subtree of the root by the maximum effect across these subtrees. Combining this with $||x_L||_2 \leq B_x$, we note from (7) that

$$\begin{aligned} \Delta_L \;\leq\;& C_\phi B_x\,||(W_1-W_1')||_2 \\ &+ C_\phi B_2 C_\rho C_g d \max_{j\in C(x_L)} \Delta_{L-1,j} \qquad (9) \\ &+ C_\phi ||(W_2-W_2')R(W_1,W_2,x_L)||_2\,. \end{aligned}$$

□

Proof of Lemma 3

Proof. Note from (9) that in order for the change in embedding of the root (due to a small change in weights) to be small, we require that the last term in (9) is small. Toward that goal we bound the norm of permutation-invariant aggregation at the root node. Specifically, we note that

$$\begin{aligned} &||R(W_1,W_2,x_L)||_2 \\ =\;& \left|\left|\rho\Big(\sum_{j\in C(x_L)} g(T_{L-1,j}(W_1,W_2))\Big)\right|\right|_2 \\ =\;& \left|\left|\rho\Big(\sum_{j\in C(x_L)} g(T_{L-1,j}(W_1,W_2))\Big)-\rho(0)\right|\right|_2 \\ \leq\;& C_\rho\left|\left|\sum_{j\in C(x_L)} g(T_{L-1,j}(W_1,W_2))\right|\right|_2 \\ \leq\;& C_\rho \sum_{j\in C(x_L)} \left|\left|g(T_{L-1,j}(W_1,W_2))-g(0)\right|\right|_2 \\ \leq\;& C_\rho C_g \sum_{j\in C(x_L)} \left|\left|T_{L-1,j}(W_1,W_2)\right|\right|_2. \\ \leq\;& C_\rho C_g d \max_{j\in C(x_L)} \left|\left|T_{L-1,j}(W_1,W_2)\right|\right|_2, \qquad (10) \end{aligned}$$

where the norm of the embedding produced by children j of the root using weights W_1 and W_2 is given by

$$\begin{aligned} &\left|\left|T_{L-1,j}(W_1,W_2)\right|\right|_2 \\ =\;& \left|\left|\phi(W_1x_{L-1,j}+W_2R(W_1,W_2,x_{L-1,j}))\right|\right|_2 \\ =\;& \left|\left|\phi(W_1x_{L-1,j}+W_2R(W_1,W_2,x_{L-1,j}))-\phi(0)\right|\right|_2 \\ \leq\;& C_\phi\left|\left|W_1x_{L-1,j}+W_2R(W_1,W_2,x_{L-1,j})\right|\right|_2 \\ \leq\;& C_\phi\left|\left|W_1x_{L-1,j}\right|\right|_2 + C_\phi\left|\left|W_2R(W_1,W_2,x_{L-1,j})\right|\right|_2 \\ \leq\;& C_\phi B_1B_x + C_\phi B_2\left|\left|R(W_1,W_2,x_{L-1,j})\right|\right|_2. \qquad (11) \end{aligned}$$

Also, since $||\phi(x)||_\infty \leq b$ for all $x\in\mathbb{R}^r$ (by our assumption), and $||\phi(x)||_2 \leq \sqrt{r}||\phi(x)||_\infty$, we obtain

$$\left|\left|T_{L-1,j}(W_1,W_2)\right|\right|_2 \;\leq\; b\sqrt{r}\,. \qquad (12)$$

Combining (10) and (11), we get the recursive relationship

$$\begin{aligned} &||R(W_1,W_2,x_L)||_2 \\ \leq\;& C_\rho C_g C_\phi B_1B_x d \\ +\;& C_\rho C_g C_\phi B_2 d \max_{j\in C(x_L)} \left|\left|R(W_1,W_2,x_{L-1,j})\right|\right|_2 \\ \leq\;& C_\rho C_g C_\phi B_1B_x d \sum_{\ell=0}^{L-1} (C_\rho C_g C_\phi B_2 d)^\ell \\ =\;& C_\rho C_g C_\phi B_1B_x d\,\frac{(\mathcal{C}d)^L-1}{\mathcal{C}d-1}\,. \qquad (13) \end{aligned}$$

On the other hand, combining (10) and (11), we get

$$||R(W_1,W_2,x_L)||_2 \;\leq\; bdC_\rho C_g\sqrt{r}\,. \qquad (14)$$

Taken together, (13) and (14) yield $||R(W_1,W_2,x_L)||_2$

$$\leq\; C_\rho C_g d \min\left\{b\sqrt{r}, C_\phi B_1B_x\frac{(\mathcal{C}d)^L-1}{\mathcal{C}d-1}\right\}. \qquad (15)$$

□

Proof of Lemma 4

Proof. Using the results from Lemma 2 and 3, we will simplify the bound on Δ_L, i.e., the change in embedding due to a change in weights. We will then bound the change in probability (that the tree label is 1) Λ_L in terms of Δ_L, when we change not only the weights from (W_1,W_2) to W_1',W_2' but also the local classifier parameters from β to β' (where β and β' are chosen from a bounded norm family). We show these steps below.

Plugging the bound on $\overline{R} \triangleq ||R(W_1, W_2, x_L)||_2$ from Lemma 3 in Lemma 2, we get

$$\begin{aligned}\Delta_L &\leq C_\phi B_x \, ||W_1 - W_1'||_2 \\ &+ \mathcal{C}d \max_{j \in C(x_L)} \Delta_{L-1,j} \\ &+ C_\phi ||W_2 - W_2'||_2 \overline{R} \, .\end{aligned}$$

Expanding the recursion, we note that

$$\Delta_L \leq M B_x \, ||W_1 - W_1'||_2 + M\overline{R}||W_2 - W_2'||_2 \, , \quad (16)$$

where

$$M = C_\phi \frac{(\mathcal{C}d)^L - 1}{\mathcal{C}d - 1} \, . \quad (17)$$

Since $||A||_2 \leq ||A||_F$ for every matrix A, we have

$$\Delta_L \leq M B_x \, ||W_1 - W_1'||_F + M\overline{R}||W_2 - W_2'||_F \, . \quad (18)$$

Now since sigmoid is 1-Lipschitz, we have

$$\begin{aligned}\Lambda_L &= |\psi(\beta^\top T_L(W_1, W_2)) - \psi(\beta'^\top T_L(W_1', W_2'))| \\ &\leq |\beta^\top T_L(W_1, W_2) - \beta'^\top T_L(W_1, W_2)| \\ &+ |\beta'^\top T_L(W_1, W_2) - \beta'^\top T_L(W_1', W_2')| \\ &\leq ||\beta - \beta'||_2 \, ||T_L(W_1, W_2)||_2 + B_\beta \Delta_L \\ &\leq ||\beta - \beta'||_2 \underbrace{(C_\phi B_1 B_x + C_\phi B_2 \overline{R})}_{Z} + B_\beta \Delta_L\end{aligned} \quad (19)$$

using (11) and (15). □

Proof of Lemma 5

Proof. Building on results from Lemmas 2-4, we will now show that the change in probability Λ_L can be bounded by ϵ, using a covering of size P, where P depends on ϵ. Moreover, we show that $\log P$ grows as $\mathcal{O}\left(\log\left(\frac{1}{\epsilon}\right)\right)$ for sufficiently small values of ϵ. That is, we can ensure Λ_L is small by using a small covering.

We begin by noting that we can find a covering $\mathcal{C}\left(\beta, \frac{\epsilon}{3Z_\ell}, ||\cdot||_2\right)$ of size

$$\mathcal{N}\left(\beta, \frac{\epsilon}{3Z_\ell}, ||\cdot||_2\right) \leq \left(1 + \frac{6ZB_\beta}{\epsilon}\right)^r \, .$$

Thus, for any specified ϵ, we can ensure that Λ_L is at most ϵ by finding matrix coverings $\mathcal{C}\left(W_1, \frac{\epsilon}{3MB_xB_\beta}, ||\cdot||_F\right)$ and $\mathcal{C}\left(W_2, \frac{\epsilon}{3M\overline{R}B_\beta}, ||\cdot||_F\right)$. Using Lemma 8 from (Chen et al., 2020a), we obtain the corresponding bounds on their covering number. Specifically,

$$\mathcal{N}\left(W_1, \frac{\epsilon}{3MB_xB_\beta}, ||\cdot||_F\right) \leq \left(1 + \frac{6MB_xB_\beta B_1\sqrt{r}}{\epsilon}\right)^{r^2},$$

$$\mathcal{N}\left(W_2, \frac{\epsilon}{3M\overline{R}B_\beta}, ||\cdot||_F\right) \leq \left(1 + \frac{6M\overline{R}B_\beta B_2\sqrt{r}}{\epsilon}\right)^{r^2}.$$

The product of all the covering numbers is bounded by

$$P = \left(1 + \frac{6B_\beta \max\{Z, M\sqrt{r}\max\{B_xB_1, \overline{R}B_2\}\}}{\epsilon}\right)^{2r^2+r}.$$

Therefore, the class $\mathcal{B}(L, d, r, \beta, B_1, B_2, B_x)$ that maps a tree-structured input to the probability that the corresponding tree label is 1 can be approximated to within ϵ by a covering of size P. Moreover, when

$$\epsilon < 6B_\beta \max\{Z, M\sqrt{r}\max\{B_xB_1, \overline{R}B_2\}\},$$

we obtain that $\log P$ is at most

$$3r^2 \log\left(\frac{12B_\beta \max\{Z, M\sqrt{r}\max\{B_xB_1, \overline{R}B_2\}\}}{\epsilon}\right) .$$

□

Proof of Proposition 7

Proof. We are now ready to prove our generalization bound. Specifically, we invoke a specific form of Dudley's entropy integral to bound the empirical Rademacher complexity $\hat{\mathcal{R}}_\mathcal{T}(\mathcal{J}_\gamma)$ via our result on covering from Lemma 5, where recall that $\mathcal{J}_\gamma$ maps each tree-label pair (t, y) to margin loss $\text{loss}_\gamma(-\tau(f_c(t;\Theta), y))$.

It is straightforward to show that p is 2-Lipschitz in its first argument, and $loss_\gamma$ is $\frac{1}{\gamma}$-Lipschitz. Therefore, we can approximate the class $\mathcal{I}$ that maps (t, y) to $\tau(f_c(t;\Theta), y)$ by finding an $\frac{\epsilon}{2}$-cover of $\mathcal{B}$. Now, note that $\mathcal{I}$ takes values in the interval $[-e, e]$, where

$$e = ||u||_2 \, ||T_L(W_1, W_2)||_2 \leq B_\beta Z \, .$$

Using Lemma A.5. in (Bartlett et al., 2017), we obtain that

$$\hat{\mathcal{R}}_\mathcal{T}(\mathcal{I}) \leq \inf_{\alpha > 0}\left(\frac{4\alpha}{\sqrt{m}} + \frac{12}{m}\int_\alpha^{2e\sqrt{m}} \sqrt{\log \mathcal{N}(\mathcal{I}, \epsilon, ||\cdot||)}d\epsilon\right)$$

where, using Lemma 5, we have

$$\begin{aligned}&\int_\alpha^{2e\sqrt{m}} \sqrt{\log \mathcal{N}(\mathcal{I}, \epsilon, ||\cdot||)}d\epsilon \\ &\leq \int_\alpha^{2e\sqrt{m}} \sqrt{\log \mathcal{N}(\mathcal{B}, \frac{\epsilon}{2}, dist(\cdot,\cdot))}d\epsilon \\ &\leq \int_\alpha^{2e\sqrt{m}} \sqrt{\log U} \quad \leq 2e\sqrt{m}\sqrt{\log U} \quad = 2B_\beta Z\sqrt{m \log U}\end{aligned}$$

with $dist$ being the combination of $||\cdot||_2$ and $||\cdot||_F$ norms used to obtain covering of size P in Lemma 5, and $\log U$ is

$$3r^2 \log\left(\frac{24B_\beta \max\{Z, M\sqrt{r}\max\{B_x B_1, \overline{R}B_2\}\}}{\alpha}\right) .$$

Setting $\alpha = \sqrt{\frac{1}{m}}$, we immediately get

$$\hat{\mathcal{R}}_{\mathcal{T}}(\mathcal{I}) \leq \frac{4}{m} + \frac{24B_\beta Z}{\sqrt{m}}\sqrt{3r^2 \log Q} ,$$

where

$$Q = 24B_\beta\sqrt{m}\max\{Z, M\sqrt{r}\max\{B_x B_1, \overline{R}B_2\}\} .$$

We finally bound the complexity of class $\hat{\mathcal{R}}_{\mathcal{T}}(\mathcal{J}_\gamma)$ by noting that $loss_\gamma$ is $\frac{1}{\gamma}$-Lipschitz, and invoking Talagrand's lemma (Mohri et al., 2012):

$$\hat{\mathcal{R}}_{\mathcal{T}}(\mathcal{J}_\gamma) \leq \frac{\hat{\mathcal{R}}_{\mathcal{T}}(\mathcal{I})}{\gamma} \leq \frac{4}{\gamma m} + \frac{24rB_\beta Z}{\gamma\sqrt{m}}\sqrt{3\log Q} .$$

□

Proof of Proposition 8

Proof. We first convey some intuition. Suppose $|X| < 8$, and we assign a distinct index $z(x) \in \{1, 2, \ldots, 8\}$ to each message $x \in X$. Then, we can map each x to $10^{-z(x)}$, i.e., obtain a decimal expansion which may be viewed as a one-hot vector representation of at most 10 digits. We would reserve a separate block of 10 digits for each port. This would allow us to disentangle the coupling between messages and their corresponding ports. Specifically, since the ports are all distinct, we can *shift* the digits in expansion of x to the right by dividing by 10^p, where p is the port number of x. This allows us to represent each (x, p) pair uniquely.

Formally, since $\mathcal{X}$ is countable, there exists a mapping $Z : \mathcal{X} \mapsto \mathbb{N}$ from $x \in \mathcal{X}$ to natural numbers. Since X has bounded cardinality, we know the existence of some $N \in \mathbb{N}$ such that $|X| < N$ for all X. Define $k = 10^{\lceil \log_{10} N \rceil}$. We define function f in the proposition as $f(x) = k^{-Z(x)}$. We also take function g in proposition to be $g(p) = 10^{-kN(p-1)}$. That is, we express the function h as $h((x_1, p_1), \ldots, (x_{|P|}, p_{|P|})) = \sum_{i=1}^{|P|} g(p_i)f(x_i)$. □