A Statistical Characterization of Attentions in Graph Neural Networks

Mufei Li
AWS Shanghai AI Lab
limufe@amazon.com

Hao Zhang *
Chongqing University of Posts and Telecommunications
sufeidechabei@gmail.com

Xingjian Shi
Unaffiliated
xshiab@connect.ust.hk

Minjie Wang
New York University/AWS
wanminji@amazon.com

Zheng Zhang
AWS Shanghai AI Lab
zhaz@amazon.com

Abstract

Does attention matter and, if so, when? While attention mechanism in Graph Attention Networks (GATs) was partially motivated to deal with unseen data, our empirical study on both inductive and transductive learning suggests that datasets have a much stronger influence. Independent of learning setting, attentions degenerate to simple averaging for all three citation networks, whereas they behave strikingly different in the protein-protein interaction dataset: nodes attend to different neighbors per head, and get more focused in deeper layers. Consequently, attention distributions become telltale features of the datasets themselves.

1 Introduction

The modeling of graphs has become an active research topic in deep learning (Bronstein et al., 2017). Dozens of neural network models have been developed recently (Scarselli et al., 2009; Bruna et al., 2014; Henaff et al., 2015; Duvenaud et al., 2015; Niepert et al., 2016; Defferrard et al., 2016), now collectively referred to as graph neural networks (GNNs). Many of them have achieved state-of-the-art performance on tasks like node classification (Kipf & Welling, 2017; Hamilton et al., 2017), link prediction (Zhang & Chen, 2018) and graph classification (Xu et al., 2019).

Recently, Velickovic et al. (2018) proposed the graph attention networks (GATs) which integrate multi-head self-attention into node feature update. Several extensions and improvements have been developed since then (Thekumparampil et al., 2018; Zhang et al., 2018; Monti et al., 2018; Svoboda et al., 2019; Trivedi et al., 2019). While these attention-based GNNs have achieved the state-of-the-art results, a thorough understanding of graph attention is yet to be achieved.

In this paper, we develop a paradigm for a systematic study of the attentions in GNNs. With extensive experiments, our findings suggest that the attentions learned by GATs are highly dataset-dependent. The attention distributions across heads and layers are near uniform for all citation networks (Cora, Citeseer and Pubmed) while they get more concentrated over layers on the protein-protein interaction dataset, with different heads have learned significantly different attentions. Furthermore, we perform a meta graph classification experiment to distinguish graphs with attention based features. A high test accuracy is achieved with interesting visualization results.

2 Background

Let $G$ be an undirected graph with node set $\mathcal{V}$, where each node $i \in \mathcal{V}$ has a feature $h_i^0 \in \mathbb{R}^{n_0}$. In a wide class of GNNs (Kipf & Welling, 2017; Hamilton et al., 2017; Velickovic et al., 2018), the basic feature update function for node $i \in \mathcal{V}$ at the $l+1$-th layer takes the form of

$$h_i^{l+1} = \sigma\left( \sum_{j \in \mathcal{N}(i)} \alpha_{i,j}^{l+1} W^{l+1} h_j^l \right),$$

\*Work done at New York University Shanghai
where $\sigma$ is an activation function, $\mathcal{N}(i)$ is a set containing $i$ and its neighbors, $\alpha_{i,j}^{l+1,k} \in \mathbb{R}$ is the attention weight of node $j$ in updating the feature of node $i$, $W^{l+1} \in \mathbb{R}^{n_{l+1} \times n_l}$ is the projection matrix, and $h_i^l, h_i^{l+1}$ are correspondingly node features after the $l$-th and the $l+1$-th layer.

**Graph Convolutional Networks (GCN)** [Kipf & Welling 2017] and the mean variant of **GraphSAGE** [Hamilton et al., 2017] uses static attention weights given by $\frac{1}{\sqrt{|\mathcal{N}(i)|}} \frac{1}{\sqrt{|\mathcal{N}(j)|}}$ and $\frac{1}{|\mathcal{N}(i)|}$.

**GAT** [Velickovic et al., 2018] uses a parameterized subnetwork to output the attention weights $\alpha_{i,j}$’s. Rather than using a single attention head as in Eqn. 1, GAT aggregates the outputs of multiple heads:

$$
\alpha_{i,j}^{l+1,k} = \gamma_k(h_i^l, h_j^l, \{h_m | m \in \mathcal{N}(i)\}), \quad h_i^{l+1} = \sigma \left( \left\| \sum_{j \in \mathcal{N}(i)} \alpha_{i,j}^{l+1,k} W^{l+1,k} h_j^l \right\| \right),
$$

where $\gamma_k$ is the subnetwork that outputs the attention weights of the $k$-th head, $\alpha_{i,j}^{l+1,k}$ and $W^{l+1,k}$ are the attention weights and projection matrix of the $k$-th head, and $\|$ means joint concatenation.

### Tasks and Datasets
We consider the node classification task with two settings: transductive learning and inductive learning. In the transductive learning setting, the model can access the features of all nodes in the graph. However, only a fraction of the nodes are labeled and the model is asked to predict the missing labels. In the inductive learning setting, we have two mutually exclusive sets of nodes separately for training and test. The model is trained only on the features and labels of the training set and is asked to predict the labels of the nodes in the test set. As in Velickovic et al. (2018), we consider the following four datasets – citation networks Cora, Citeseer [Sen et al., 2008], Pubmed [Namata et al., 2012] and protein-protein interaction dataset (PPI) Zitnik & Leskovec, 2017.

### 3 Methodology

The introduction of multi-head attention into multi-layer GNNs poses four questions. **Q1**: In the GAT model, all nodes have different attention distributions on their incoming edges. How should we characterize the overall statistics of these learned attention distributions? **Q2**: For a single node, multiple attention distributions are calculated by different architectural components such as heads and layers. How do these attention distributions differ across different heads and layers? **Q3**: How does the choice of the dataset and the learning setting affect the learned attentions? **Q4**: Is the statistics of the learned attention related to the intrinsic properties of the graph?

To answer **Q1**, we propose multiple metrics to characterize the overall statistics of a collection of attention distributions. To alleviate the impact of randomness in the training phase, we train GAT with multiple seeds and calculate the metrics using all the learned attentions. We also visualize some metrics to intuitively understand the attentions learned by GAT. For **Q2**, to investigate the layer-wise differences, we examine the characteristics of the attentions generated by different layers using the aforementioned method; to investigate the head-wise variance, we define a metric that is based on the statistical distance of two distributions. To answer **Q3**, we run experiments to see how varying the dataset and the learning setting impacts the learned attentions. Previous works [Kipf & Welling, 2017], [Hamilton et al., 2017], [Velickovic et al., 2018] only perform transductive learning on the citation networks and inductive learning on **PPI**. To fill in the gap, we perform transductive learning on **PPI** and inductive learning on the citation networks, of which the data processing strategy is explained in the appendix [A.1]. We show that the learning task has little impact on the overall metric statistics. To answer **Q4**, we propose a new task called **Meta Graph Classification** which asks the model to distinguish the type of the graphs by the characteristics of the attention distributions.

### 4 Experiments

All experiments are performed based on the code released by Velickovic et al. (2018). For transductive learning on citation networks and inductive learning on **PPI**, we use the best hyperparameters reported. For the rest two experiments, a hyperparameter search is performed on the validation set. Unless mentioned otherwise, the results are based on 100 random runs for transductive learning on citation networks and 10 random runs for rest combinations of dataset and learning setting.
Figure 1: Entropy histogram plots for attention and datasets variants. For the GAT cases, we plot the attentions by the first head of the first layer in trained models. The results are merged across multiple runs. The PPI results are based on training with about 79% nodes.

The main findings of our experiments are the following. First, the deciding factor for the attention is the dataset itself. The statistics of it in PPI differs significantly from that of the citation networks. The attention distributions in all citation networks are near uniform, regardless of the heads and layers. For PPI, the distributions get sharper with deeper layers. Furthermore, different heads of a layer sharply attend to different neighbors. Lastly, the meta graph classification experiment suggests that our proposed metrics are potentially telltale features of the nature of the graphs.

Impact of Datasets To investigate the impact of the dataset on attentions, we first run transductive learning experiments on all datasets and examine the learned attentions. For PPI, the micro $F_1$ score are separately $0.544 \pm 0.022$ and $0.904 \pm 0.005$ for training with about 5% and 79% of the nodes. The analysis of attentions is given in the following paragraphs.

Overall Results We refer to the simple averaging in the mean variant of GraphSAGE as mean attention and the symmetric normalization weights in GCN as GCN attention. They are determined solely by the graph topology, whereas the GAT attentions combine both topology and node features. For a node $i \in V$, the dispersion of its attentions over its incoming edges can be measured by entropy, i.e., $H(\{\alpha_{i,j} \mid j \in \mathcal{N}(i)\}) = - \sum_{j \in \mathcal{N}(i)} \alpha_{i,j} \log \alpha_{i,j}$. To understand the general attention dispersion over the graph(s), we use the histogram plot of attention entropies for all nodes. Figure 1 shows the entropy histogram plots for mean attentions, GCN attentions, and the learned GAT attentions on all four datasets. For GCN attentions, we first perform a normalization $\frac{\alpha_{i,j}}{\sum_{j \in \mathcal{N}(i)} \alpha_{i,j}}$ so that the attentions sum up to 1. For PPI, we merge the results from all 24 graphs. One can observe that the histogram plots of learned attentions in all citation networks are similar to those in the mean attention case and differ slightly from those in the GCN attention case. This suggests that by and large the attention weights are roughly the same for different neighbors. However, the attentions learned for PPI appear significantly different. Analogously, we examine the dispersion of Jacobian-based saliency values (Papernot et al., 2016), see appendix A.2.

Layer-wise Statistics We examine the layer-wise differences of attentions with several metrics: maximum pairwise difference, maximum attention value within the neighborhood set $\mathcal{N}(i)$, and attention on self loop. We average the metrics over all nodes and heads for a layer in each run. Then, we compute the mean and standard deviation of the averaged metrics in all runs. The results for Cora and PPI with 79% nodes for training are visualized in figure 3. A table view of the results (including the ones for the rest experiments) can be found in appendix A.3. The metrics in the PPI case suggest that attentions get more concentrated and have a sharper focus over their neighborhood with deeper layers while they change little in the cases of citation networks. At first, we suspect that such focus might be pointing to the node itself. This turns out not to be the case: the attentions on self loops on average change little across layers despite the increasingly concentrated attentions. These observations hold for both dataset split (training with 5% and 79% of the nodes), with the attention concentration being much more extreme when significantly more nodes are used for training.
Head-wise Statistics For one random run, figure 2 visualizes the attentions of a node over its neighbors in Cora and PPI, based on three heads in the last layer. We can find that different heads behave distinctively in PPI case and they are all uniform in Cora case. We further propose metrics to quantify the head-wise differences. We compute first the mean attention distributions for all heads within the same layer. The head-wise variance is then determined by the averaged $L_1$ norm of the difference between the mean distributions and the distributions for each head. The $L_1$ norm of the difference between two distributions is also known as total variation.

$$
\alpha_{i,j}^{\text{mean}} = \frac{1}{K} \sum_{k=1}^{K} \alpha_{i,j}^{k}, \ j \in \mathcal{N}(i), \quad \text{Head-wise Variance} = \frac{1}{2K} \frac{1}{|V|} \sum_{i=1}^{K} \sum_{j \in V} \sum_{j \in \mathcal{N}(i)} |\alpha_{i,j}^{k} - \alpha_{i,j}^{\text{mean}}| \quad (3)
$$

We record the head-wise variance from multiple random runs for all datasets in appendix A.4.1. For citation networks, the head-wise variance is small as all attention distributions are close to uniform distributions. For PPI, we observe significant head-wise variance which generally gets larger for deeper layers, suggesting that different heads attend to different part of the neighbors. In addition, the head-wise variance measured is more significant when more nodes are used for training.

Impact of Learning Tasks There are two factors that potentially affect the learned attentions: dataset and learning setting. The previous results suggest that the choice of dataset plays a key role. Nevertheless, we need to verify whether the choice of task induces a difference. Therefore, we further examine the attentions learned with the inductive learning setting on all datasets. For inductive learning on citation networks, the test accuracy for Cora, Citeseer and Pubmed are separately 88.32 $\pm$ 0.31%, 83.38 $\pm$ 0.22% and 87.60 $\pm$ 0.28%. The layer-wise and head-wise differences are separately recorded in appendix A.3.2 and appendix A.4.2. The statistics in the inductive learning setting have minor differences with those in the transductive learning setting. The general observation still holds: the attentions learned on PPI are much more sharper with high head-wise variance.

Meta Graph Classification Previous experiments demonstrate that the attentions learned are highly graph-dependent and their characteristics can be predicted with proper knowledge of graphs. A follow-up question is whether we can do the inverse problem, i.e., infer the graph types based on the attentions learned. Inspired by this idea, we perform graph classification with attention based features. For each dataset, we sample 120 subgraphs as in the case of inductive learning on citation networks and separately train a 3-layer GAT on them to classify the nodes for inductive learning. The mean and standard deviation of layer-wise attention metrics for all 3 layers from 10 runs are
then used as graph features. We train a logistic regression classifier using 20% of training graphs with features standardized. The test accuracy with 10 random train, test splits is 97.4 ± 1.7%. If we use only metrics from the first, second or third layer, the test accuracy is separately 95.8 ± 2.9%, 76.2 ± 3.1% and 70.9 ± 4.0%. Figure 4 shows t-SNE (van der Maaten & Hinton, 2008) visualization of the attention metrics separately for all layers, the first layer, the second layer, and the final layer. We can find that the features for citation networks are close to each other and get more indistinguishable with deeper layers. On the other hand, the features of PPI are far from those of the citation networks across all layers.

REFERENCES


### Appendix

#### A.1 Varying Learning Settings

**Transductive Learning on PPI** To perform transductive learning on PPI, we sample two mutually exclusive subsets of the nodes as the training set and validation set for each graph, leaving the rest as the test set. We experiment on two splitting settings. In the first setting, we sample about 5% nodes for training and 18% nodes for validation, similar to the splitting ratio of the transductive learning setting on Cora. In the second setting, we sample 79% nodes for training and 11% nodes for validation, similar to the case of inductive learning on PPI.

**Inductive Learning on Citation Networks** To perform inductive learning on citation networks, we first sample 120 graphs of 100 nodes for each dataset. We use a random walk based sampling algorithm described in Algorithm 1 which by the study of Leskovec & Faloutsos (2006) performs best in preserving the properties of static graphs. Separately, 60%, 20%, 20% of the graphs are used for training, validation and test.

#### A.2 Jacobian-based Saliency Values

The Jacobian-based saliency values of node $i$ with respect to its neighbor $j$ is defined as $s_{i,j}^{l+1,k} = \left| \frac{\partial h_{i,j}^{l+1,k}}{\partial h_j^l} \right|$, and may be interpreted as the “contribution” of node $j$ in updating the feature of node $i$. We normalize these values by $\sum_{j \in N(i)} s_{i,j}^{l+1,k}$ to compute the entropy.

Below we compare the entropy histogram plots of attention and saliency values in a same run for Cora and PPI. For this comparison, we use the setting of Veličković et al. (2018). The histogram plots of the entropy values for saliency look quite similar to those for attentions in the case of citation networks. Meanwhile, differences are observed for the case of PPI.

#### A.2.1 Cora

Figure 5 and 6 look basically the same as that of the mean attention. This is also the case for Citeseer and Pubmed.
Algorithm 1 Random Walk Sampling

Require: $G = (\mathcal{V}, \mathcal{E})$ the original graph, $g_{size} = 100$ the target subgraph size
1: $\text{step} = 0$
2: $\text{start} \sim \text{Unif}(\mathcal{V})$ \hspace{1cm} \triangleright \text{Uniformly choose a starting node.}
3: $\mathcal{V}_{\text{sub}} = \{\text{start}\}, \mathcal{E}_{\text{sub}} = \{(\text{start}, \text{start})\}$
4: $\text{src} = \text{start}$
5: while $|\mathcal{V}_{\text{sub}}| < g_{size}$ and $\text{step} < 100 \times g_{size}$ do
6: $\text{step} = \text{step} + 1$
7: $\text{back} \sim \text{Bernoulli}(0.15)$, \hspace{1cm} \triangleright \text{Return to the starting point with probability 0.15.}
8: if $\text{back}$ then
9: $\text{src} = \text{start}$
10: else
11: $\text{dst} \sim \text{Unif}(\{j | (\text{src}, j) \in \mathcal{E}\})$
12: $\mathcal{V}_{\text{sub}} = \mathcal{V}_{\text{sub}} \cup \{\text{dst}\}$
13: $\mathcal{E}_{\text{sub}} = \mathcal{E}_{\text{sub}} \cup \{(\text{dst}, \text{dst}), (\text{src}, \text{dst}), (\text{dst}, \text{src})\}$
14: $\text{src} = \text{dst}$
15: Return $(\mathcal{V}_{\text{sub}}, \mathcal{E}_{\text{sub}})$

---

Figure 5: Entropy histogram plot of attentions for all heads in a trained GAT.

A.2.2 PPI

Figure 7 and 8 compare the entropy histogram plot of attentions and saliency values for one graph in PPI. Different from the cases of citation networks, we do have observed clear differences between the two cases.

A.3 Layer-wise Differences

A.3.1 Transductive Learning

Table 1, 2, 3, 4, and 5 separately summarizes the layerwise-metrics for transductive learning on Cora, Citeseer, Pubmed and PPI (with two settings).
Figure 6: Entropy histogram plot of saliency values for all heads in a trained GAT.

Figure 7: Entropy histogram plot of attentions for all heads in a trained GAT.

Figure 8: Entropy histogram plot of saliency values for all heads in a trained GAT.

Table 1: Layer-wise differences for transductive learning on Cora

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.006 ± 0.001</td>
<td>0.007 ± 0.006</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.279 ± 0.000</td>
<td>0.279 ± 0.003</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.275 ± 0.000</td>
<td>0.275 ± 0.000</td>
</tr>
</tbody>
</table>

A.3.2 Inductive Learning

Table 6, 7, 8, and 9 separately summarizes the layer-wise metrics for inductive learning on Cora, Citeseer, Pubmed and PPI.
Table 2: Layer-wise differences for transductive learning on Citeseer

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.001 ± 0.000</td>
<td>0.009 ± 0.005</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.360 ± 0.000</td>
<td>0.364 ± 0.002</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.359 ± 0.000</td>
<td>0.360 ± 0.000</td>
</tr>
</tbody>
</table>

Table 3: Layer-wise differences for transductive learning on Pubmed

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.013 ± 0.003</td>
<td>0.052 ± 0.005</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.347 ± 0.002</td>
<td>0.367 ± 0.003</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.340 ± 0.000</td>
<td>0.342 ± 0.000</td>
</tr>
</tbody>
</table>

Table 4: Layer-wise differences for transductive learning on PPI with about 5% nodes for training

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.286 ± 0.111</td>
<td>0.648 ± 0.136</td>
<td>0.730 ± 0.083</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.330 ± 0.105</td>
<td>0.665 ± 0.128</td>
<td>0.741 ± 0.081</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.115 ± 0.003</td>
<td>0.167 ± 0.014</td>
<td>0.155 ± 0.039</td>
</tr>
</tbody>
</table>

Table 5: Layer-wise differences for transductive learning on PPI with about 79% nodes for training

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.435 ± 0.052</td>
<td>0.753 ± 0.049</td>
<td>0.895 ± 0.019</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.471 ± 0.049</td>
<td>0.764 ± 0.049</td>
<td>0.905 ± 0.019</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.109 ± 0.003</td>
<td>0.172 ± 0.013</td>
<td>0.173 ± 0.013</td>
</tr>
</tbody>
</table>

Table 6: Layer-wise differences for inductive learning on Cora

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.067 ± 0.023</td>
<td>0.027 ± 0.009</td>
<td>0.020 ± 0.004</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.381 ± 0.012</td>
<td>0.360 ± 0.005</td>
<td>0.356 ± 0.002</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.348 ± 0.003</td>
<td>0.342 ± 0.002</td>
<td>0.344 ± 0.000</td>
</tr>
</tbody>
</table>

Table 7: Layer-wise differences for inductive learning on Citeseer

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.111 ± 0.041</td>
<td>0.030 ± 0.010</td>
<td>0.024 ± 0.004</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.403 ± 0.023</td>
<td>0.359 ± 0.006</td>
<td>0.355 ± 0.002</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.347 ± 0.004</td>
<td>0.339 ± 0.002</td>
<td>0.341 ± 0.000</td>
</tr>
</tbody>
</table>

A.4 Head-wise Differences

A.4.1 Transductive Learning

Table [10] summarizes the head-wise variances across datasets for transductive learning.
Table 8: Layer-wise differences for inductive learning on Pubmed

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.007 ± 0.002</td>
<td>0.004 ± 0.002</td>
<td>0.008 ± 0.003</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.369 ± 0.001</td>
<td>0.367 ± 0.001</td>
<td>0.369 ± 0.001</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.365 ± 0.000</td>
<td>0.365 ± 0.000</td>
<td>0.365 ± 0.000</td>
</tr>
</tbody>
</table>

Table 9: Layer-wise differences for inductive learning on PPI

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max pairwise difference</td>
<td>0.458 ± 0.042</td>
<td>0.782 ± 0.039</td>
<td>0.917 ± 0.015</td>
</tr>
<tr>
<td>Max attention</td>
<td>0.493 ± 0.041</td>
<td>0.791 ± 0.038</td>
<td>0.927 ± 0.016</td>
</tr>
<tr>
<td>Attention on self loop</td>
<td>0.109 ± 0.005</td>
<td>0.192 ± 0.009</td>
<td>0.180 ± 0.018</td>
</tr>
</tbody>
</table>

Table 10: Head-wise variance for transductive learning

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cora</td>
<td>0.004 ± 0.001</td>
<td>0.000 ± 0.000</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Citeseer</td>
<td>0.001 ± 0.000</td>
<td>0.000 ± 0.000</td>
<td>Not applicable</td>
</tr>
<tr>
<td>Pubmed</td>
<td>0.007 ± 0.002</td>
<td>0.031 ± 0.007</td>
<td>Not applicable</td>
</tr>
<tr>
<td>PPI-setting1 (about 5% nodes for training)</td>
<td>0.297 ± 0.088</td>
<td>0.463 ± 0.089</td>
<td>0.438 ± 0.073</td>
</tr>
<tr>
<td>PPI-setting2 (about 79% nodes for training)</td>
<td>0.436 ± 0.028</td>
<td>0.559 ± 0.037</td>
<td>0.649 ± 0.021</td>
</tr>
</tbody>
</table>

Table 11: Head-wise variance for inductive learning

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Layer3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cora</td>
<td>0.038 ± 0.010</td>
<td>0.014 ± 0.004</td>
<td>0.005 ± 0.001</td>
</tr>
<tr>
<td>Citeseer</td>
<td>0.060 ± 0.020</td>
<td>0.016 ± 0.006</td>
<td>0.005 ± 0.001</td>
</tr>
<tr>
<td>Pubmed</td>
<td>0.003 ± 0.001</td>
<td>0.002 ± 0.001</td>
<td>0.004 ± 0.002</td>
</tr>
<tr>
<td>PPI</td>
<td>0.450 ± 0.019</td>
<td>0.539 ± 0.032</td>
<td>0.647 ± 0.027</td>
</tr>
</tbody>
</table>

A.4.2 Inductive Learning

Table 11 summarizes the head-wise variances across datasets for inductive learning.