Appendix for "Diffusion Improves Graph Learning"

A Graph diffusion as a polynomial filter

We want to find a direct correspondence between graph diffusion with θ_k and a polynomial filter with parameters ξ_j, i.e.

$$\sum_{j=0}^{J} \xi_j \boldsymbol{L}^j \overset{!}{=} \sum_{k=0}^{K} \theta_k \boldsymbol{T}^k. \tag{8}$$

To do so, we first expand $\boldsymbol{T} = \boldsymbol{I}_N - \boldsymbol{L}$ and use the binomial equation, i.e.

$$\begin{aligned}
\sum_{k=0}^{K} \theta_k \boldsymbol{T}^k &= \sum_{k=0}^{K} \theta_k (\boldsymbol{I}_N - \boldsymbol{L})^k = \\
&= \sum_{k=0}^{K} \theta_k \sum_{j=0}^{k} \binom{k}{j} (-1)^j \boldsymbol{I}_N^{k-j} \boldsymbol{L}^j = \\
&= \sum_{k=0}^{K} \sum_{j=0}^{k} \binom{k}{j} \theta_k (-1)^j \boldsymbol{L}^j = \\
&= \sum_{\substack{j,k \in [0,K] \\ j \leq k}} \binom{k}{j} \theta_k (-1)^j \boldsymbol{L}^j = \\
&= \sum_{j=0}^{K} \underbrace{\sum_{k=j}^{K} \binom{k}{j} \theta_k (-1)^j}_{\xi_j} \boldsymbol{L}^j,
\end{aligned} \tag{9}$$

where we recognize the coefficients ξ_j and see that we need to set $J = K$. Note that we reordered the summation indices by recognizing the triangular sum, i.e. the sum over index pairs (j, k) with $j \leq k$. The equation for conversion in the opposite direction is obtained in the same way since $\boldsymbol{L} = \boldsymbol{I}_N - \boldsymbol{T}$. To obtain a more convenient form for $K \to \infty$ we shift the summation index using $m = k - j$, i.e.

$$\xi_j = \sum_{k=j}^{K} \binom{k}{j} (-1)^j \theta_k = \sum_{m=0}^{K-j} \binom{m+j}{j} (-1)^j \theta_{m+j}. \tag{10}$$

To find corresponding coefficients for the heat kernel, we let $K \to \infty$, set $\theta_k = e^{-t} \frac{t^k}{k!}$, and use the exponential series to obtain

$$\begin{aligned}
\xi_j^{\text{HK}} &= \sum_{m=0}^{\infty} \binom{m+j}{j} (-1)^j e^{-t} \frac{t^{m+j}}{(m+j)!} = \\
&= \sum_{m=0}^{\infty} \frac{(m+j)!}{m!j!} (-1)^j e^{-t} \frac{t^{m+j}}{(m+j)!} = \\
&= e^{-t} \frac{(-t)^j}{j!} \sum_{m=0}^{\infty} \frac{t^m}{m!} = \frac{(-t)^j}{j!} e^{-t} e^t = \frac{(-t)^j}{j!}.
\end{aligned} \tag{11}$$

To obtain the coefficients for PPR, we let $K \to \infty$, set $\theta_k = \alpha(1-\alpha)^k$, and recognize the series expansion $\frac{1}{(1-x)^{j+1}} = \sum_{m=0}^{\infty} \binom{m+j}{m} x^m$, resulting in

$$\begin{aligned}
\xi_j^{\text{PPR}} &= \sum_{m=0}^{\infty} \binom{m+j}{j} (-1)^j \alpha (1-\alpha)^{m+j} = \\
&= \alpha (-1)^j (1-\alpha)^j \sum_{m=0}^{\infty} \binom{m+j}{m} (1-\alpha)^m = \\
&= \alpha (\alpha - 1)^j \frac{1}{\alpha^{j+1}} = \left(1 - \frac{1}{\alpha}\right)^j.
\end{aligned} \tag{12}$$

B Experiments

For optimizing the hyperparameters for node classification the data is split into a development and a test set. The development set contains 1500 nodes for all datasets but for COAUTHOR CS, where it contains 5000 nodes. All remaining nodes are part of the test set and only used once for testing. The development set is split into a training set containing 20 nodes per class and a validation set with the remaining nodes. For every run the accuracy is determined using 100 different random splits of the development set using fixed seeds. Different seeds are used for validation and test splits. Early stopping patience is set to 100 epochs with a maximum limit of 10000 epochs, which is never reached. The patience is reset after an increase in accuracy on the validation set. For the test runs we select the hyperparameter configurations that showed the highest average accuracy on the validation splits.

We use the same development set for optimizing the hyperparameters for clustering. The test set is only once for generating test results. Clustering results are averaged over 20 randomly initialized runs.

Confidence intervals are calculated by bootstrapping the accuracy results from 100 or 20 runs, respectively, with 1000 samples. All implementations for node classification as well as DGI are based on PyTorch [58] and PyTorch Geometric [21]. The remaining experiments are based on NumPy [72], SciPy [30], graph-tool [60], and gensim [20]. For k-means clustering we use the implementation by scikit-learn [59]. All datasets are included in PyTorch Geometric, available at `https://github.com/rusty1s/pytorch_geometric`. Experiments using PyTorch are run on Nvidia GPUs using CUDA and the remaining experiments are run on Intel CPUs.

For all experiments the largest connected component of the graph is selected. Dropout probability is set to $p = 0.5$ for all experiments and performed after every application of the activation function. PPR preprocessing is done with $\alpha \in [0.05, 0.30]$, heat kernel preprocessing with $t \in [1, 10]$. For top-k matrix sparsification k is set to either 64 or 128 and for ϵ-thresholding ϵ is chosen from $[0.00001, 0.01]$. We do not choose ϵ directly but rather calculate which ϵ corresponds to a chosen average degree. For node classification we use the Adam optimizer with a learning rate of 0.01. The hidden dimension of GNNs is kept fixed at 64 with the exception of ARMA, where the dimensionality of a single stack is chosen from 16 or 32. For ARMA, up to three stacks and two layers are tested. GCN and GAT are run with up to 4 layers, JK and GIN with up to six layers. L_2-regularization is performed on the parameters of the first layer of every model with $\lambda_{L_1} \in [0.001, 10]$. Unsupervised models use a node embedding dimension of 128. DGI uses the Adam optimizer with a learning rate of 0.001. For a full list of final hyperparameters per model, diffusion, and dataset see Sec. B.3.

B.1 Datasets

Table 1: Dataset statistics.

Dataset	Type	Classes	Features	Nodes	Edges	Label rate
CORA	Citation	7	1433	2485	5069	0.056
CITESEER	Citation	6	3703	2120	3679	0.057
PUBMED	Citation	3	500	19 717	44 324	0.003
COAUTHOR CS	Co-author	15	6805	18 333	81 894	0.016
AMZ COMP	Co-purchase	10	767	13 381	245 778	0.015
AMZ PHOTOS	Co-purchase	8	745	7487	119 043	0.021

B.2 Results

To support our claim of achieving state-of-the-art node classification performance we also include results (and hyperparameters) of APPNP, which has been shown to be the current state of the art for semi-supervised node classification and uses graph diffusion internally [21, 23].

Table 2: Average accuracy (%) on CORA with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	81.71 ± 0.26	83.48 ± 0.22	83.58 ± 0.23	82.93 ± 0.23
GAT	80.10 ± 0.34	81.54 ± 0.25	81.60 ± 0.25	81.32 ± 0.22
JK	82.14 ± 0.24	83.69 ± 0.29	83.78 ± 0.22	83.43 ± 0.21
GIN	73.96 ± 0.46	76.54 ± 0.63	78.74 ± 0.44	75.94 ± 0.45
ARMA	81.62 ± 0.24	83.32 ± 0.22	83.81 ± 0.21	83.24 ± 0.22
APPNP	$\mathbf{83.83 \pm 0.23}$	-	-	-
DCSBM	59.75 ± 1.59	64.63 ± 2.60	68.52 ± 1.47	-
Spectral	29.29 ± 1.03	35.16 ± 2.96	34.03 ± 2.01	-
DeepWalk	68.67 ± 1.01	68.76 ± 0.67	69.42 ± 0.07	-
DGI	54.29 ± 1.21	67.71 ± 1.69	$\mathbf{69.61 \pm 1.73}$	-

Table 3: Average accuracy (%) on CITESEER with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	72.02 ± 0.31	73.22 ± 0.27	$\mathbf{73.35 \pm 0.27}$	71.58 ± 0.31
GAT	69.52 ± 0.32	70.25 ± 0.34	68.50 ± 0.21	68.68 ± 0.22
JK	70.34 ± 0.38	72.38 ± 0.27	72.24 ± 0.31	71.11 ± 0.33
GIN	61.09 ± 0.58	62.82 ± 0.50	64.07 ± 0.48	61.46 ± 0.51
ARMA	70.84 ± 0.32	71.90 ± 0.33	72.28 ± 0.29	71.45 ± 0.31
APPNP	72.76 ± 0.25	-	-	-
DCSBM	46.70 ± 2.18	56.81 ± 1.21	57.14 ± 1.40	-
Spectral	27.02 ± 0.57	29.61 ± 1.29	29.26 ± 1.46	-
DeepWalk	55.33 ± 1.05	66.05 ± 0.56	65.81 ± 0.16	-
DGI	54.62 ± 2.28	71.58 ± 0.94	$\mathbf{72.42 \pm 0.39}$	-

Table 4: Average accuracy (%) on PUBMED with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	78.23 ± 0.40	79.62 ± 0.36	78.72 ± 0.37	77.46 ± 0.36
GAT	76.32 ± 0.47	77.78 ± 0.34	76.66 ± 0.32	75.98 ± 0.33
JK	78.47 ± 0.36	$\mathbf{79.95 \pm 0.28}$	79.22 ± 0.32	78.01 ± 0.41
GIN	72.38 ± 0.63	74.16 ± 0.62	73.62 ± 0.63	68.14 ± 0.80
ARMA	77.14 ± 0.36	79.64 ± 0.35	78.85 ± 0.36	77.32 ± 0.37
APPNP	79.78 ± 0.33	-	-	-
DCSBM	46.64 ± 1.85	67.38 ± 1.45	64.51 ± 1.75	-
Spectral	37.97 ± 0.02	49.28 ± 3.08	48.05 ± 2.69	-
DeepWalk	70.77 ± 0.14	$\mathbf{71.36 \pm 0.14}$	69.96 ± 0.12	-
DGI	49.96 ± 2.21	65.94 ± 0.23	66.52 ± 0.35	-

Table 5: Average accuracy (%) on COAUTHOR CS with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	91.83 ± 0.08	92.79 ± 0.07	$\mathbf{93.01 \pm 0.07}$	92.28 ± 0.06
GAT	90.89 ± 0.13	89.82 ± 0.10	91.33 ± 0.07	88.29 ± 0.06
JK	91.11 ± 0.09	92.40 ± 0.08	92.41 ± 0.07	91.68 ± 0.08
ARMA	91.32 ± 0.08	92.32 ± 0.09	92.63 ± 0.08	91.03 ± 0.09
APPNP	92.08 ± 0.07	-	-	-
DCSBM	57.70 ± 1.52	63.70 ± 0.93	61.71 ± 1.15	-
Spectral	24.74 ± 2.28	50.47 ± 3.20	55.27 ± 3.00	-
DeepWalk	61.26 ± 0.91	63.77 ± 1.28	$\mathbf{65.29 \pm 1.40}$	-
DGI	57.52 ± 2.63	62.84 ± 1.84	63.79 ± 1.89	-

Table 6: Average accuracy (%) on AMZ COMP with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	84.75 ± 0.23	$\mathbf{86.77 \pm 0.21}$	86.04 ± 0.24	85.73 ± 0.23
GAT	45.37 ± 4.20	86.68 ± 0.26	85.37 ± 0.33	86.55 ± 0.26
JK	83.33 ± 0.27	86.51 ± 0.26	85.66 ± 0.30	84.40 ± 0.32
GIN	55.44 ± 0.83	81.11 ± 0.62	75.08 ± 1.20	56.52 ± 1.65
ARMA	84.36 ± 0.26	86.09 ± 0.27	84.92 ± 0.26	84.92 ± 0.29
APPNP	81.72 ± 0.25	-	-	-
DCSBM	44.61 ± 0.77	55.80 ± 1.29	$\mathbf{57.92 \pm 2.25}$	-
Spectral	40.39 ± 1.11	50.89 ± 3.05	52.62 ± 2.14	-
DeepWalk	55.61 ± 0.25	56.29 ± 0.50	55.05 ± 0.98	-
DGI	30.84 ± 1.96	37.27 ± 1.21	36.81 ± 1.12	-

Table 7: Average accuracy (%) on AMZ PHOTO with bootstrap-estimated 95% confidence levels.

Model	No diffusion	Heat	PPR	AdaDIF
GCN	92.08 ± 0.20	92.82 ± 0.23	92.20 ± 0.22	92.37 ± 0.22
GAT	53.40 ± 5.49	91.86 ± 0.20	90.89 ± 0.27	91.65 ± 0.20
JK	91.07 ± 0.26	$\mathbf{92.93 \pm 0.21}$	92.37 ± 0.22	92.34 ± 0.22
GIN	68.34 ± 1.16	87.24 ± 0.65	83.41 ± 0.82	75.37 ± 0.86
ARMA	91.41 ± 0.22	92.05 ± 0.24	91.09 ± 0.24	90.38 ± 0.28
APPNP	91.42 ± 0.26	-	-	-
DCSBM	66.30 ± 1.70	67.13 ± 2.49	64.28 ± 1.81	-
Spectral	28.15 ± 0.81	49.86 ± 2.06	53.65 ± 3.22	-
DeepWalk	78.82 ± 0.85	$\mathbf{79.26 \pm 0.09}$	78.73 ± 0.10	-
DGI	40.09 ± 2.14	49.02 ± 1.78	51.34 ± 1.96	-

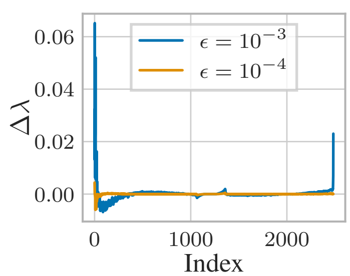

Figure 11: Close-up of difference caused by sparsification (Fig. 2b). Primarily the lowest and highest eigenvalues of the Laplacian are affected.

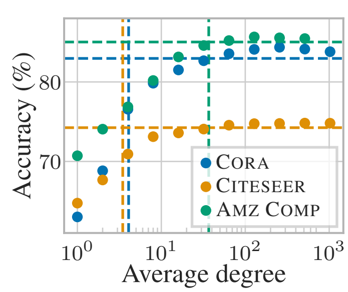

Figure 12: GCN+GDC accuracy (using PPR and sparsification by threshold ϵ). Lines indicate original accuracy and degree. GDC surpasses the original accuracy at around the same degree independent of dataset. Sparsification can improve accuracy.

B.3 Hyperparameters

Table 8: Hyperparameters for GCN obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	λ_{L_2}	Learning rate	Dropout	Hidden dimension	Hidden depth
-	CORA	-	-	-	-	0.06	0.01	0.5	64	1
	CITESEER					10.0				
	PUBMED					0.03				
	COAUTHOR CS					0.06				
	AMZ COMP					0.03				
	AMZ PHOTO					0.03				
Heat	CORA	-	5	-	0.0001	0.09	0.01	0.5	64	1
	CITESEER		4	-	0.0009	10.0				1
	PUBMED		3	-	0.0001	0.04				1
	COAUTHOR CS		1	64	-	0.08				1
	AMZ COMP		5	-	0.0010	0.07				1
	AMZ PHOTO		3	-	0.0001	0.08				2
PPR	CORA	0.05	-	128	-	0.10	0.01	0.5	64	1
	CITESEER	0.10			0.0009	10.0				
	PUBMED	0.10		64		0.06				
	COAUTHOR CS	0.10		64		0.03				
	AMZ COMP	0.10		64		0.04				
	AMZ PHOTO	0.15		64		0.03				
AdaDIF	CORA	-	-	128	-	0.08	0.01	0.5	64	1
	CITESEER			128		0.08				1
	PUBMED			128		0.01				2
	COAUTHOR CS			64		0.03				1
	AMZ COMP			64		0.02				1
	AMZ PHOTO			64		0.02				1

Table 9: Hyperparameters for GAT obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	λ_{L_2}	Learning rate	Dropout	Hidden dimension	Hidden depth
	CORA					0.06				1
	CITESEER					0.06				1
-	PUBMED	-	-	-	-	0.03	0.01	0.5	64	2
	COAUTHOR CS					0.00				2
	AMZ COMP					0.09				1
	AMZ PHOTO					0.08				1
	CORA				0.0010	0.04				1
	CITESEER				0.0010	0.08				1
Heat	PUBMED	-	1	-	0.0005	0.02	0.01	0.5	64	2
	COAUTHOR CS				0.0005	0.03				1
	AMZ COMP				0.0005	0.01				1
	AMZ PHOTO				0.0005	0.01				1
	CORA				0.0050	0.08				1
	CITESEER				0.0005	0.10				1
PPR	PUBMED	0.10	-	-	0.0005	0.00	0.01	0.5	64	2
	COAUTHOR CS				0.0005	0.00				1
	AMZ COMP				0.0005	0.03				1
	AMZ PHOTO				0.0005	0.07				2
	CORA			-	0.0010	0.04				1
	CITESEER			-	0.0005	0.04				1
AdaDIF	PUBMED	-	-	128	-	0.01	0.01	0.5	64	2
	COAUTHOR CS			64	-	0.02				1
	AMZ COMP			64	-	0.02				1
	AMZ PHOTO			64	-	0.02				1

Table 10: Hyperparameters for JK obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	λ_{L_2}	Learning rate	Dropout	Aggregation	Hidden dimension	Hidden depth
	CORA					0.04					3
	CITESEER					1.00					4
-	PUBMED	-	-	-	-	0.05	0.01	0.5	Concatenation	64	2
	COAUTHOR CS					0.02					2
	AMZ COMP					0.03					2
	AMZ PHOTO					0.03					2
	CORA		5	-	0.0001	0.09					
	CITESEER		4	-	0.0009	1.00					
Heat	PUBMED	-	3	-	0.0001	0.09	0.01	0.5	Concatenation	64	2
	COAUTHOR CS		1	64	-	0.03					
	AMZ COMP		5	-	0.0010	0.07					
	AMZ PHOTO		3	-	0.0005	0.07					
	CORA	0.05		128		0.10					
	CITESEER	0.2			0.0009	1.00					
PPR	PUBMED	0.10	-	64	-	0.02	0.01	0.5	Concatenation	64	2
	COAUTHOR CS	0.10		64		0.03					
	AMZ COMP	0.10		64		0.04					
	AMZ PHOTO	0.15		64		0.03					
	CORA			128		0.05					2
	CITESEER			128		0.08					2
AdaDIF	PUBMED	-	-	128	-	0.01	0.01	0.5	Concatenation	64	3
	COAUTHOR CS			64		0.02					2
	AMZ COMP			64		0.03					2
	AMZ PHOTO			64		0.02					2

Table 11: Hyperparameters for GIN obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	λ_{L_2}	Learning rate	Dropout	Aggregation	Hidden dimension	Hidden depth
	CORA					0.09					4
	CITESEER					0.10					4
-	PUBMED	-	-	-	-	0.08	0.01	0.5	Sum	64	4
	AMZ COMP					0.01					5
	AMZ PHOTO					0.01					4
	CORA		3	-	0.0001	0.07					5
	CITESEER		8	-	0.0009	0.01					4
Heat	PUBMED	-	3	-	0.0010	0.02	0.01	0.5	Sum	64	5
	AMZ COMP		3	64	-	0.00					4
	AMZ PHOTO		3	64	-	0.00					4
	CORA	0.05		128		0.01					4
	CITESEER	0.05			0.0009	0.01					4
PPR	PUBMED	0.10	-	64	-	0.01	0.01	0.5	Sum	64	5
	AMZ COMP	0.10		64		0.04					4
	AMZ PHOTO	0.10		64		0.04					4
	CORA			128		0.02					3
	CITESEER			128		0.05					4
AdaDIF	PUBMED	-	-	64	-	0.03	0.01	0.5	Sum	64	5
	AMZ COMP			64		0.02					4
	AMZ PHOTO			64		0.02					4

Table 12: Hyperparameters for ARMA obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	λ_{L_2}	Learning rate	Dropout	ARMA layers	ARMA stacks	Hidden dimension	Hidden depth
	CORA					0.04				3		
	CITESEER					0.08				3		
-	PUBMED	-	-	-	-	0.00	0.01	0.5	1	2	16	1
	COAUTHOR CS					0.02				2		
	AMZ COMP					0.01				3		
	AMZ PHOTO					0.01				3		
	CORA		5	64	-	0.08				2		
	CITESEER		5	64	-	0.08				3		
Heat	PUBMED	-	3	-	0.0001	0.00	0.01	0.5	1	2	16	1
	COAUTHOR CS		1	64	-	0.01				3		
	AMZ COMP		5	64	-	0.04				3		
	AMZ PHOTO		3	64	-	0.04				2		
	CORA	0.10		128		0.05				3	16	
	CITESEER	0.15		128		0.08				3	16	
PPR	PUBMED	0.10	-	64	-	0.01	0.01	0.5	1	3	16	1
	COAUTHOR CS	0.10		64		0.01				2	16	
	AMZ COMP	0.10		128		0.06				2	32	
	AMZ PHOTO	0.15		128		0.06				2	32	
	CORA			128		0.05				2		
	CITESEER			128		0.09				3		
AdaDIF	PUBMED	-	-	64	-	0.01	0.01	0.5	1	2	16	1
	COAUTHOR CS			64		0.03				2		
	AMZ COMP			64		0.01				3		
	AMZ PHOTO			64		0.01				2		

Table 13: Hyperparameters for APPNP obtained by grid and random search.

Dataset name	α	k	λ_{L_2}	Learning rate	Dropout	Hidden dimension	Hidden depth
CORA	0.10		0.09				
CITESEER	0.10		1.00				
PUBMED	0.10	10	0.02	0.01	0.5	64	1
COAUTHOR CS	0.15		0.01				
AMZ COMP	0.10		0.06				
AMZ PHOTO	0.10		0.05				

Table 14: Hyperparameters for DCSBM obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	Number of blocks
-	CORA	-	-	-	-	7
	CITESEER					6
	PUBMED					3
	COAUTHOR CS					15
	AMZ COMP					10
	AMZ PHOTO					8
Heat	CORA	-	5	-	0.0010	7
	CITESEER		1	64	-	6
	PUBMED		3	64	-	3
	COAUTHOR CS		5	-	0.0010	15
	AMZ COMP		3	-	0.0010	10
	AMZ PHOTO		3	-	0.0010	8
PPR	CORA	0.05	-	-	0.0010	7
	CITESEER	0.05		64	-	6
	PUBMED	0.10		-	0.0010	3
	COAUTHOR CS	0.05		64	-	15
	AMZ COMP	0.05		-	0.0010	10
	AMZ PHOTO	0.10		64	-	8

Table 15: Hyperparameters for spectral clustering obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	Embedding dimension
-	CORA	-	-	-	-	128
	CITESEER					
	PUBMED					
	COAUTHOR CS					
	AMZ COMP					
	AMZ PHOTO					
Heat	CORA	-	5	-	0.0010	128
	CITESEER		5	-	0.0010	
	PUBMED		5	64	-	
	COAUTHOR CS		5	-	0.0010	
	AMZ COMP		1	64	-	
	AMZ PHOTO		5	64	-	
PPR	CORA	0.10	-	-	0.0010	128
	CITESEER	0.05		-	0.0010	
	PUBMED	0.15		-	0.0010	
	COAUTHOR CS	0.05		64	-	
	AMZ COMP	0.05		64	-	
	AMZ PHOTO	0.15		64	-	

Table 16: Hyperparameters for DeepWalk obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	Walks per node	Embedding dimension	Walk length
-	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	-	-	-	-	10	128	64
Heat	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	-	5 1 1 5 3 3	- 64 - - - -	0.0010 - 0.0010 0.0010 0.0010 0.0010	10	128	64
PPR	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	0.05 0.05 0.15 0.10 0.05 0.15	-	- - 64 64 - -	0.0010 0.0010 - - 0.0010 0.0010	10	128	64

Table 17: Hyperparameters for DGI obtained by grid and random search.

Diffusion	Dataset name	α	t	k	ϵ	Learning rate	Encoder	Embedding dimension
-	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	-	-	-	-	0.001	GCN	128
Heat	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	-	1 5 5 5 1 1	- - 64 64 - -	0.0001 0.0001 - - 0.0001 0.0001	0.001	GCN	128
PPR	CORA CITESEER PUBMED COAUTHOR CS AMZ COMP AMZ PHOTO	0.15 0.10 0.15 0.15 0.30 0.30	-	-	0.0001 0.0001 0.0010 0.0010 0.0010 0.0010	0.001	GCN	128