

A Dual-Graph Attention-Based Approach for Identifying Distribution Network Topology

Guoqiang Lin, Siyan Liu, Danni Shi, Xuan Wang, Shiliang Liu Artificial Intelligence on Electric Power System State Grid Corporation Joint Laboratory (State Grid Smart Grid Research Institute Co.,Ltd.)

Beijing 102209, China

The distribution network needs to obtain the correct topology in time during the generation at work. Here our array problems of the case of the generation of the intervent operating to the section measurement problems of the during processing of the model in the section measurement problems of the during processing of the model in the section measurement problems of the during processing of the model in the section measurement problems of the during processing of the model in the section measurement problems of the during processing of the model in the section measurement problems of the during processing of the model in the section measurement problems of the during processing during the during the during the during processing during the d	01	Topology identification framework						
TI-DGA modelExperimentsConvolutional layer: $Z_{\mu} = c(GNN(0,A) + A^{2}))$ Called a function graph pooling layer: $Z = c(D^{-1}\overline{A}D^{-1}Sa_{MD})$ TABLE II COMPARISON OF ALGORITHM PERFORMANCE ON IEEE33 NODESAttention graph pooling layer: $Z = c(D^{-1}\overline{A}D^{-1}Sa_{MD})$ TABLE II COMPARISON OF ALGORITHM PERFORMANCE ON IEEE33 NODESConcaterial network with two iden layers.TABLE II COMPARISON OF ALGORITHM PERFORMANCE ON 0.921ON 0.921O.762O.777CONNULA)GNN(P,A+A*A)ON IEEE33 NODESConcateriateTI-DGAO.936O.827ConcateriateON IEEE33 NODESConcateriatePreRec ON IEEE33 NODESConcateriatePreRec ON IEEE33 NODESConcateriatePreRec ON IEEE33 NODESConcateriatePreRec ON IEEE33 NODESConcateriatePreConcateriatePreRec ON IEEE33 NODESConcateriatePreConcateriatePreConcateriatePreConcateriatePreConcateriatePre <th col<="" th=""><th colspan="2"> The distribution network needs to obtain the correct topology in time during the operation to adjust the control strategy and ensure the safe operation of the distribution network. For the current problems of frequent changes of distribution network topology and difficulties in obtaining topology structure in real time, in this paper , we propose a distribution network topology identification method based on dual graph attention mechanism (TI-DGA), which firstly obtains the attention score by inputting different features and weight matrices in the graph convolution layer, and then uses a pooling mechanism to select top-k node features as the features of the whole graph to achieve the classification purpose. Contributions: we proposed a dual graph attention mechanism method TI-DGA, and the model can be suitable for application to topology feature learning and association feature learning of many different types of data. the model framework proposed in this paper is suitable for the application of online and offline patterns of distribution network topology identification. compared with other algorithms, the algorithm proposed in this paper has better accuracy on IEEE33 nodes and IEEE57 nodes. </th><th colspan="5"> Distribution network topology identification includes the following processes: Data Acquisition: the section measurement data, node voltage amplitude U, node injected active power P, and the corresponding topology labels are used as the initial samples. Data Processing: If the model is trained directly with the original data, it will increase the training difficulty and affect the topology recognition effect, so we normalize the voltage data. Missing data in the test set need to be filled to reduce their impact. In this paper, the kNN algorithm is used to interpolate the missing data. Offline Training: The filtered features from the above two steps are used as the input to the model, and the optimization and updating of the model parameters are achieved by iterations of the minimization error function, which consists of the actual and desired outputs of the model. Online Application: In this step, the collected measurement data of the current time section of the distribution network is input into the trained learning model, so that the operational topology of the current section of the distribution network can be output in real time. </th></th>	<th colspan="2"> The distribution network needs to obtain the correct topology in time during the operation to adjust the control strategy and ensure the safe operation of the distribution network. For the current problems of frequent changes of distribution network topology and difficulties in obtaining topology structure in real time, in this paper , we propose a distribution network topology identification method based on dual graph attention mechanism (TI-DGA), which firstly obtains the attention score by inputting different features and weight matrices in the graph convolution layer, and then uses a pooling mechanism to select top-k node features as the features of the whole graph to achieve the classification purpose. Contributions: we proposed a dual graph attention mechanism method TI-DGA, and the model can be suitable for application to topology feature learning and association feature learning of many different types of data. the model framework proposed in this paper is suitable for the application of online and offline patterns of distribution network topology identification. compared with other algorithms, the algorithm proposed in this paper has better accuracy on IEEE33 nodes and IEEE57 nodes. </th> <th colspan="5"> Distribution network topology identification includes the following processes: Data Acquisition: the section measurement data, node voltage amplitude U, node injected active power P, and the corresponding topology labels are used as the initial samples. Data Processing: If the model is trained directly with the original data, it will increase the training difficulty and affect the topology recognition effect, so we normalize the voltage data. Missing data in the test set need to be filled to reduce their impact. In this paper, the kNN algorithm is used to interpolate the missing data. Offline Training: The filtered features from the above two steps are used as the input to the model, and the optimization and updating of the model parameters are achieved by iterations of the minimization error function, which consists of the actual and desired outputs of the model. Online Application: In this step, the collected measurement data of the current time section of the distribution network is input into the trained learning model, so that the operational topology of the current section of the distribution network can be output in real time. </th>	 The distribution network needs to obtain the correct topology in time during the operation to adjust the control strategy and ensure the safe operation of the distribution network. For the current problems of frequent changes of distribution network topology and difficulties in obtaining topology structure in real time, in this paper , we propose a distribution network topology identification method based on dual graph attention mechanism (TI-DGA), which firstly obtains the attention score by inputting different features and weight matrices in the graph convolution layer, and then uses a pooling mechanism to select top-k node features as the features of the whole graph to achieve the classification purpose. Contributions: we proposed a dual graph attention mechanism method TI-DGA, and the model can be suitable for application to topology feature learning and association feature learning of many different types of data. the model framework proposed in this paper is suitable for the application of online and offline patterns of distribution network topology identification. compared with other algorithms, the algorithm proposed in this paper has better accuracy on IEEE33 nodes and IEEE57 nodes. 		 Distribution network topology identification includes the following processes: Data Acquisition: the section measurement data, node voltage amplitude U, node injected active power P, and the corresponding topology labels are used as the initial samples. Data Processing: If the model is trained directly with the original data, it will increase the training difficulty and affect the topology recognition effect, so we normalize the voltage data. Missing data in the test set need to be filled to reduce their impact. In this paper, the kNN algorithm is used to interpolate the missing data. Offline Training: The filtered features from the above two steps are used as the input to the model, and the optimization and updating of the model parameters are achieved by iterations of the minimization error function, which consists of the actual and desired outputs of the model. Online Application: In this step, the collected measurement data of the current time section of the distribution network is input into the trained learning model, so that the operational topology of the current section of the distribution network can be output in real time. 				
Convolutional layer: $Z_{v} = \sigma(GNN(U, A))$ $Z_{r} = \sigma(GNN(U, A))$ $Z_{r} = \sigma(GNN(U, A))$ $Z_{r} = \sigma(GNN(U, A) + A^2)$ Readout layer: $s = \frac{1}{h} \sum_{i=1}^{r} X_{i}(I) \max_{i=1}^{max} X_{i}$ MLP layer: a feedforward fully conceted neumal network with two hidden layers.TABLE I COMPARISON OF ALGORITHM PERFORMANCE ON IEEE33 NODES $Z = \sigma(D^{-1}XD^{-1}XO_{min})$ $Idx = top_rrank_{(2/kM)}$. $Z_{max} = Z_{idx}$ MLP layer: a feedforward fully conceted neumal network with two hidden layers. $Algorithms$ Acc Pre Rec $GNN(U,A)$ $GNN(P,A+A^*A)$ A^*A A^*A A^*B Acc Pre Rec $GNN(U,A)$ $GNN(P,A+A^*A)$ $GNN(P,A+A^*A)$ A^*A Acc Pre Rec $GNN(P,A+A^*A)$ $GNN(P,A+A^*A)$ Acc Pre Rec Acc Pre Rec Acc Pre Rec $GNN(P,A+A^*A)$ Acc Pre Rec Acc Acc Pre Rec	TI-DGA model		Experiments					
$\begin{aligned} \lambda_{y} = \sigma(\text{GNN}(P, A + A^{3})) & s = \frac{1}{h} \sum_{k=1}^{h} x_{k}^{(1)} \max x_{k} \\ \text{LightGBM} & 0.844 & 0.649 & 0.673 \\ \text{LightGBM} & 0.844 & 0.649 & 0.673 \\ \text{CBOOST} & 0.932 & 0.752 & 0.757 \\ \text{CNN} & 0.972 & 0.853 & 0.894 \\ \text{GONN}(J, A) & \text{idden layers.} \\ \text{idx} = top_rank_{(2 k V)}.Z_{mask} = Z_{idx} \\ \hline \\ GNN(U,A) & \text{GNN}(P,A+A^{*}A) & \text{GNN}(P,A+A^{*}A) \\ \hline \\ \hline \\ GNN(U,A) & \text{GNN}(P,A+A^{*}A) & \text{GNN}(P,A+A^{*}A) \\ \hline \\ \hline \\ \\ \hline \\ \\ Graph Pooling & \text{Graph Pooling} \\ \hline \\ \\ \hline \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ $	Convolutional layer:	Readout layer:	TABLE I COMPARISON OF ALGORITHM PERFORMANCE ON IEEE33 NODES					
$ \begin{aligned} \begin{aligned} \lambda_{p} = \sigma(GNN(P, A + A^{2})) & L_{p} = L_{p} L_$	$Z_U = \sigma(GNN(U, A))$	$s = \frac{1}{2} \sum_{i=1}^{N} r_{i} ll \max r_{i}$	Algorith	ns A	cc]	Pre	Rec	
Attention graph pooling layer: $Idx = co(\overline{D}^{-1}\overline{A}\overline{D}^{-1}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{A}\overline{A}\overline{D}_{-1}^{-1}\overline{A}\overline{A}\overline{A}\overline{A}\overline{A}\overline{A}\overline{A}\overline{A}\overline{A}}\overline{A}A$	$Z_{P} = \sigma(GNN(P, A + A^{2}))$	$N = \sum_{i=1}^{N} x_i \prod_{i=1}^{N} x_i \prod_{i=1}^$	LightGB	M 0.8	364 0	.649	0.673	
$\begin{aligned} \mathbf{X} = \sigma(\mathbf{\bar{D}}^{-1}\mathbf{\bar{X}}\mathbf{D}^{-1}\mathbf{\bar{X}}\mathbf{Q}_{-1}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{Q}_{-1}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{D}_{-1}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf{\bar{X}}\mathbf$	Attention graph pooling layer	MI P layer: a feedforward fully	XGBoos	t 0.9	932 0	.762	0.757	
$Z = \alpha(\overline{D}^{-1}\overline{A}\overline{D}^{-1}\overline{X}\overline{D}_{3R})$ $Z = \alpha(\overline{D}^{-1}\overline{A}\overline{D}^{-1}\overline{X}\overline{D}_{3R})$ $Idden layers.$ $Idden layers.$ $GNN(U,A)$ $for N(U,A)$ $GNN(U,A)$ $GNN(P,A+A^*A)$ $GNN(P,A+A^*A)$ $GNN(P,A+A^*A)$ $GON(P,A+A^*A)$	Attention graph pooling layer.	connected neural network with two	CNN	0.9	972 0	.853	0.894	
$idx = top_rank_{(2, k n]}, Z_{mask} = Z_{idx}$ $GNN(U,A)$ $for an (U,A)$ $GNN(U,A)$ G	$\mathbf{Z} = \sigma \left(\widetilde{\mathbf{D}}^{-\frac{1}{2}} \widetilde{\mathbf{A}} \widetilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \Theta_{\text{att}} \right)$	hidden lavers	GCN	0.9	980 0	.874	0.925	
$ \begin{array}{c} I = LQGA & 0.994 & 0.911 & 0.940 \\ \hline TABLE II COMPARISON OF ALCORNTHM PERFORMANCE \\ ON IEEE57 NODES \\ \hline TABLE II COMPARISON OF ALCORNTHM PERFORMANCE \\ ON IEEE57 NODES \\ \hline TABLE II COMPARISON OF ALCORNTHM PERFORMANCE \\ ON IEEE57 NODES \\ \hline TABLE II COMPARISON OF ACC VALUES ON IEEE53 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE53 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE53 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE53 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE \\ \hline TABLE III ALCORITHM COMP$	idr = ton rank(atture) Z = Z	indicit lay cas.	SAGPoo	1 0.9	989 0	.893	0.936	
GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(D,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) Goncatenate GCncatenate Graph Pooling Graph Pooling Graph Pooling Graph Pooling Graph Pooling Graph Pooling MLP Readout Readout Readout TABLE II ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEST NOES WITH ADDED NOISE TABLE III ALGORITHM 0015 SAGPool 0.98	$ux = cop_r unk(z, kN), z_{mask} = z_i$	dx	TI-DGA	. 0.9	994 0	.911	0.940	
Algorithms Acc Pre Rec IghtIGBM 0.812 0.578 0.603 XGBoost 0.919 0.715 0.692 CNN 0.965 0.825 0.811 GNN(U,A) GNN(P,A+A*A) 0.905 0.825 0.811 GCN 0.9975 0.892 0.919 SAGPool 0.983 0.887 0.929 T-DGA 0.988 0.906 0.931 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE33 NODES WITH ADDED NOISE TVE=0.05 TVE=1.1 S % % % % % Graph Pooling Free adout Readout TVE=0.01 TVE=0.05 TVE=1.1 S % % % % % % Graph Pooling Free adout TVE=0.01 TVE=0.05 TVE=1.1 S % % % % % GRON 0.972 0.974 0.680 0.825 CNN 0.902 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON <	GNN(U.A)	TABLE II COMPARISON OF ALGORITHM PERFORMANCE ON IEEE57 NODES						
Image: Concatenate in the problem Image:			Algorith	ms A	Acc	Pre	Rec	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $			LightGB	M 0.	.812 0	.578	0.603	
GNN(U,A) GNN(P,A+A*A) GNN(U,A) GNN(P,A+A*A) GON 0.975 0.825 0.811 GON 0.975 0.892 0.919 SAGPool 0.988 0.906 0.931 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE33 NODES WITH ADDED NOISE IEEE33 NODES WITH ADDED NOISE Algorithm TVE=0.01 TVE=0.05 TVE=0.5 S % % % V % % % Keadout Readout Readout 0.991 0.874 0.809 0.836 TI-DGA 0.992 0.931 0.846 SAGPool 0.984 0.972 0.935 0.886 TI-DGA 0.992 0.931 0.836 0.872 0.886 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE IEEE57 NODES WITH ADDED NOISE 0.836 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE IEEE57 NODES WITH ADDED NOISE 0.837 CNN 0.977 0.948 0.892 0.835 0.846 SAGPool 0.981 0.930 0.873			XGBoos	st 0.	.919 0	.715	0.692	
GNN(U,A) GNN(P,A+A*A) GOR (Concatenate) GOR (Concatenate) Graph Pooling Graph Pooling Graph Pooling Graph Pooling Graph Pooling Graph Pooling Readout Readout TH-DGA 0.983 Optimized (Concatenate) NUP Graph Pooling Graph Pooling Graph Pooling Graph Pooling Concatenate NUP Readout Readout TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEEB3 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE CNN 0.977 0.948 0.952 0.792 XGBoost 0.921 0.887 0.892 0.832 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.952 0.793 CNN 0.965 0.930 0.887 0.887			CNN	0	.965 0	.825	0.811	
SAGPool 0.983 0.887 0.929 TI-DGA 0.988 0.906 0.931 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE33 NODES WITH ADDED NOISE Algorithm TVE=0.01 TVE=0.05 TVE=1 s % % % LightGBM 0.848 0.792 0.724 0.657 XGBoost 0.919 0.874 0.809 0.838 GCN 0.976 0.923 0.912 0.846 SAGPool 0.992 0.972 0.935 0.809 GCN 0.976 0.953 0.912 0.846 SAGPool 0.992 0.921 0.935 0.868 TI-DGA 0.992 0.923 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TVE=0.5 TVE=1 S % % % MLP IghtGBM 0.817 0.767 0.681 0.592 XGBoost 0.921 0.825 0.930	GNN(U.A)	GNN(P.A+A*A)	GCN	0.	.975 0	.892	0.919	
Image: Concatenate for the concatenatenatenatenate for			SAGPoo		.983 (0.887	0.929	
Concatenate Concatenate Graph Pooling Graph Pooling Graph Pooling Graph Pooling Readout Readout TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE33 NODES WITH ADDED NOISE XGBoost 0.919 0.874 CON 0.972 0.912 0.724 0.6072 0.912 CON 0.972 0.984 0.972 0.984 0.972 0.984 0.930 0.981 0.930 0.981 0.930 MLP TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE XGBoost 0.992 MLP MLP MLP XGBoost 0.992 MLP XGBoost 0.992 XGBoost 0.992 0.881 CNN 0.965 0.930 0.873 XGBoost 0.992 0.892 0.852 XGBoost 0.991 0.982 0.933 XGBoost 0.921 0.892 0.857			11-DG/	N 0.	.988 (.906	0.931	
Graph Pooling Graph Pooling TVE=0.01 TVE=0.05 TVE=0.5 TVE=1 S % % % % % LightGBM 0.848 0.792 0.724 0.657 XGBoost 0.919 0.874 0.809 0.725 CNN 0.976 0.953 0.912 0.846 SAGPool 0.984 0.972 0.935 0.868 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840	Concatenate		TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE33 NODES WITH ADDED NOISE					
S % % % % % Graph Pooling Graph Pooling LightGBM 0.848 0.792 0.724 0.657 LightGBM 0.848 0.899 0.874 0.809 0.725 CNN 0.972 0.956 0.899 0.838 GCN 0.976 0.953 0.912 0.846 SAGPool 0.984 0.972 0.935 0.868 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE CNN 0.921 0.892 0.852 0.798 XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.950 0.936 0.857	Ļ	Ļ	Algorithm	TVE=0.01	TVE=0.05	TVE=0.5	TVE=1	
Graph Pooling Graph Pooling Graph Pooling 0.848 0.792 0.724 0.657 XGBoost 0.9172 0.956 0.899 0.838 GCN 0.976 0.953 0.912 0.846 SAGPool 0.994 0.972 0.935 0.868 TI-DGA 0.992 0.981 0.930 0.872 Algorithm TVE=0.01 TVE=0.05 TVE=0.5 TVE=1 S % % % % LightGBM 0.817 0.767 0.681 0.592 XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840			S LiebtCDM	70	<u>%</u> 0	<u> </u>	<u>70</u>	
Readout Readout Readout 0.819 0.874 0.809 0.725 CNN 0.972 0.956 0.899 0.838 GCN 0.976 0.972 0.935 0.912 0.846 SAGBoost 0.972 0.935 0.912 0.868 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE TVE=1 \$ \$ % <td>Graph Pooling</td> <td>Graph Pooling</td> <td>LightGBM</td> <td>0.848</td> <td>0.792</td> <td>0.724</td> <td>0.657</td>	Graph Pooling	Graph Pooling	LightGBM	0.848	0.792	0.724	0.657	
Readout Readout Readout 0.572 0.535 0.899 0.838 GCN 0.972 0.953 0.912 0.846 SGPool 0.984 0.972 0.935 0.868 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEES7 NODES WITH ADDED NOISE IEEES7 NODES WITH ADDED NOISE MLP MLP S % % % KGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.892 0.834 SAGPool 0.981 0.952 0.917 0.840			CNIN	0.919	0.874	0.809	0.725	
Readout Readout 0.770 0.733 0.712 0.840 SAGPool 0.984 0.972 0.935 0.868 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE MLP Sagorithm TVE=0.01 TVE=0.05 TVE=0.5 TVE=1 Sagorithm Sagorithm 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 0.592 XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840 TI-DGA 0.984 0.950 0.936 0.857			GCN	0.972	0.950	0.099	0.846	
Readout Readout 0.904 0.904 0.912 0.930 0.872 TI-DGA 0.992 0.981 0.930 0.872 TABLE III ALGORITHM COMPARISON OF ACC VALUES ON IEEE57 NODES WITH ADDED NOISE MLP S % % % KGBoost 0.921 0.802 0.873 0.815 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGROOI 0.981 0.952 0.917 0.840	⊥ ⊥	Ţ	SAGPool	0.984	0.955	0.912	0.868	
Readout Readout Iterative 0.501			TI-DGA	0.907	0.972	0.930	0.872	
Image: Mile	Readout	TABLE III ALGORITHM COMPARISON OF ACC VALUES ON						
Algorithm TVE=0.01 TVE=0.5 TVE=1 s % % % % LightGBM 0.817 0.767 0.681 0.592 XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840 TI-DGA 0.984 0.950 0.936 0.857		\frown	IEEE57 NODES WITH ADDED NOISE					
LightGBM 0.817 0.767 0.681 0.592 XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840	└── ▶ (•		Algorithm	TVE=0.01 %	TVE=0.05 %	TVE=0.5 %	TVE=1	
MLP XGBoost 0.921 0.892 0.852 0.798 CNN 0.965 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840		Ţ	LightGBM	0.817	0.767	0.681	0.592	
MLP CNN 0.926 0.930 0.873 0.815 GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840 TI-DGA 0.984 0.950 0.936 0.857		V	XGBoost	0.921	0.892	0.852	0.798	
GCN 0.977 0.948 0.893 0.834 SAGPool 0.981 0.952 0.917 0.840 TI-DGA 0.984 0.950 0.936 0.857		MLP	CNN	0.965	0.930	0.873	0.815	
SAGPool 0.981 0.952 0.917 0.840 TI-DGA 0.984 0.950 0.936 0.857			GCN	0.977	0.948	0.893	0.834	
TI-DGA 0.984 0.950 0.936 0.857			SAGPool	0.981	0.952	0.917	0.840	
			TI-DGA	0.984	0.950	0.936	0.857	