NetGAN: Generating Graphs via Random Walks

Aleksandar Bojchevski Oleksandr Shchur

Daniel Zügner Stephan Günnemann

Presenter: Yevgeny Tkach

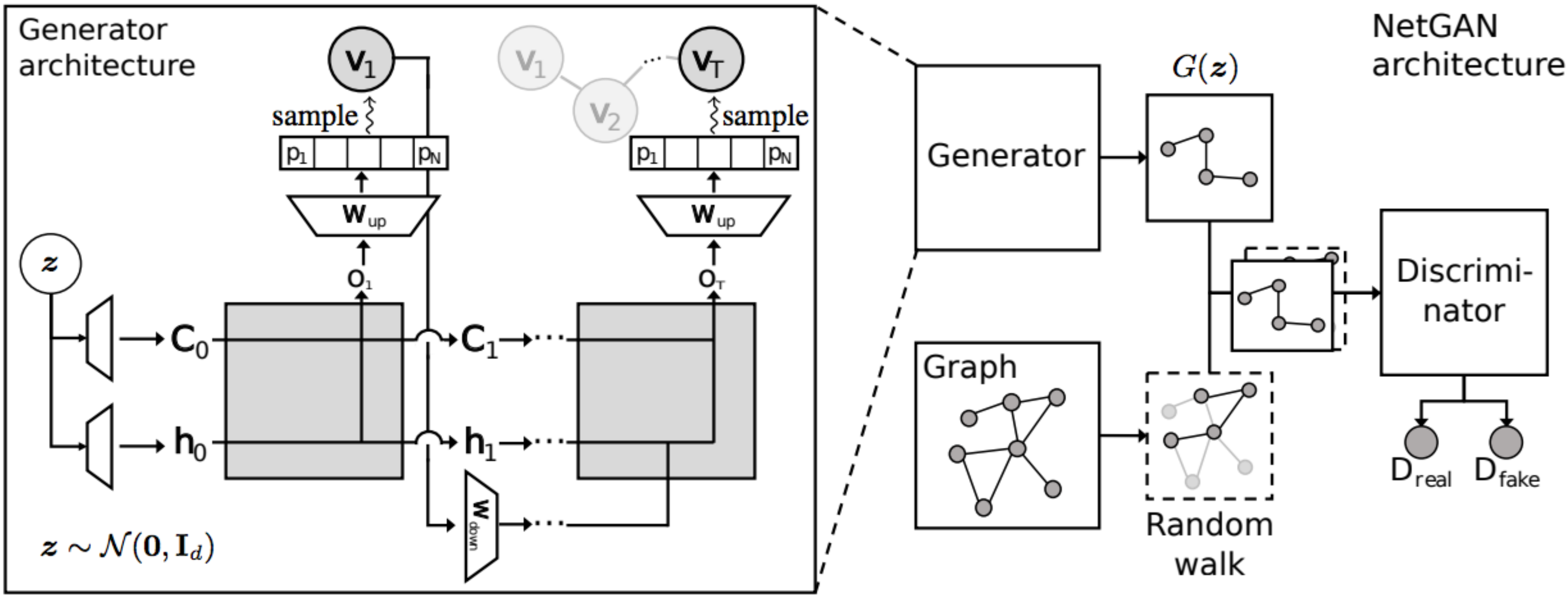
2019 Spring @

https://qdata.github.io/deep2Read/

Executive Summary

- Graph generation via random-walks on the graph.
- The task is a sub-graph generation from a single given graph/network.
- generalization of the node2vec approach to GAN. A sub product is indeed a node embedding matrix.
-

NetGAN



(a) Generator architecture

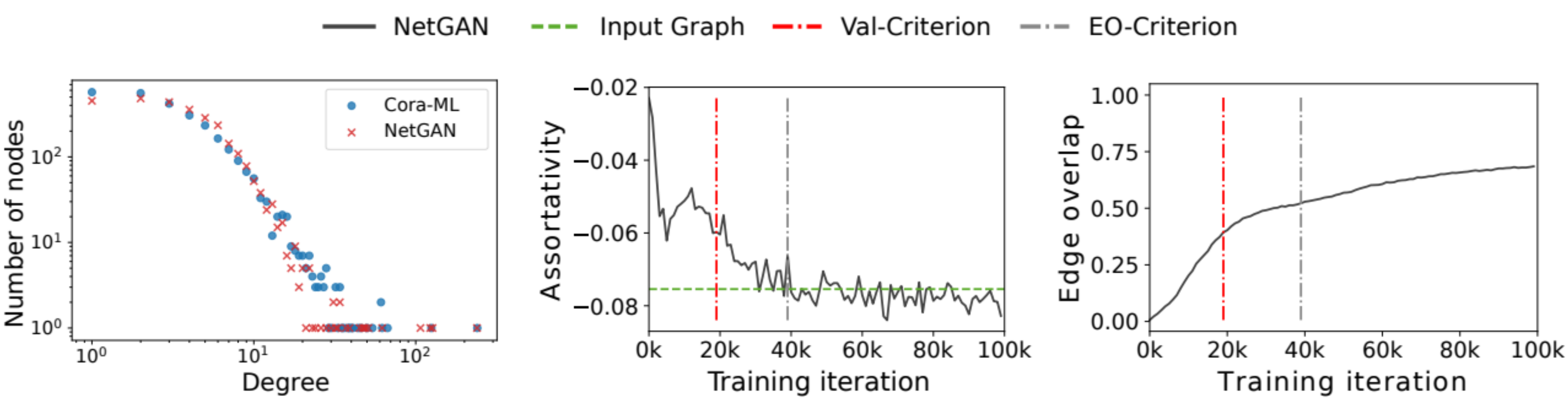
(b) NetGAN architecture

Experiments

- **Network generation -** compare network statistics to the ground truth dataset
- **Link prediction -** 10% and additional 5% of the edges are removed from the graph, while preserving connectivity for validation and test sets respectively. On those edges the algorithm is evaluated on a link prediction task using AUC and Average Precision (AP).

Experiment results

Graph		Max. degree	Assort-ativity	Triangle count	Power law exp.	Inter-comm. unity density	Intra-comm. unity density	Cluster-ing coeff.	Charac. path len.	Average rank
CORA-ML		240	-0.075	2,814	1.860	4.3e-4	1.7e-3	2.73e-3	5.61	
Conf. model	(1% EO)	*	-0.030	322	*	1.6e-3	2.8e-4	3.00e-4	4.38	7.50
Conf. model	(52% EO)	*	-0.051	626	*	9.8e-4	9.9e-4	6.10e-4	4.46	5.83
DC-SBM	(11% EO)	165	-0.052	1,403	1.814	6.7e-4	1.2e-3	3.30e-3	5.12	3.36
ERGM	(56% EO)	243	-0.077	2,293	1.786	6.9e-4	1.2e-3	2.17e-3	4.59	2.88
BTER	(2.2% EO)	199	0.033	3,060	1.787	1.0e-3	7.5e-4	4.62e-3	4.59	4.75
VGAE	(0.3% EO)	13	-0.009	14	1.674	1.4e-3	3.2e-4	1.17e-3	5.28	5.88
NetGAN VAL	(39% EO)	199	-0.060	1,410	1.773	6.5e-4	1.3e-3	2.33e-3	5.17	3.00
NetGAN EO	(52% EO)	233	-0.066	1,588	1.793	6.0e-4	1.4e-3	2.44e-3	5.20	1.75



(a) Degree distribution

(b) Assortativity over training iterations

(c) Edge overlap (EO) over training iterations

Figure 3: Properties of graphs generated by NetGAN trained on CORA-ML.

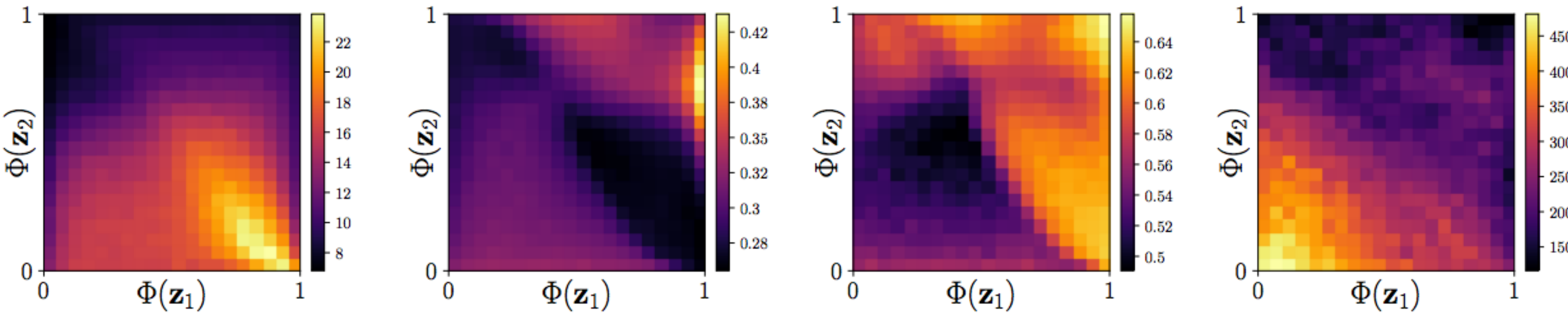
Experiment results

Table 3: Link prediction performance (in %).

Method	CORA-ML		CORA		CITESEER		DBLP		PUBMED		POLBLOGS	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP
Adamic/Adar	92.16	85.43	93.00	86.18	88.69	77.82	91.13	82.48	84.98	70.14	85.43	92.16
DC-SBM	**96.03**	95.15	98.01	97.45	94.77	93.13	**97.05**	96.57	**96.76**	95.64	95.46	**94.93**
node2vec	92.19	91.76	**98.52**	**98.36**	95.29	94.58	96.41	96.36	96.49	95.97	85.10	83.54
VGAE	95.79	**96.30**	97.59	97.93	95.11	96.31	96.38	**96.93**	94.50	**96.00**	93.73	94.12
NetGAN (500K)	94.00	92.32	82.31	68.47	95.18	91.93	82.45	70.28	87.39	76.55	95.06	94.61
NetGAN (100M)	95.19	95.24	84.82	88.04	**96.30**	**96.89**	86.61	89.21	93.41	94.59	**95.51**	94.83

Further Analysis

- The authors further demonstrate that the latent space is incorporating information about high level characteristics of the graph, by considering a 2-dimensional noise vector down from a bivariate standard normal distribution.



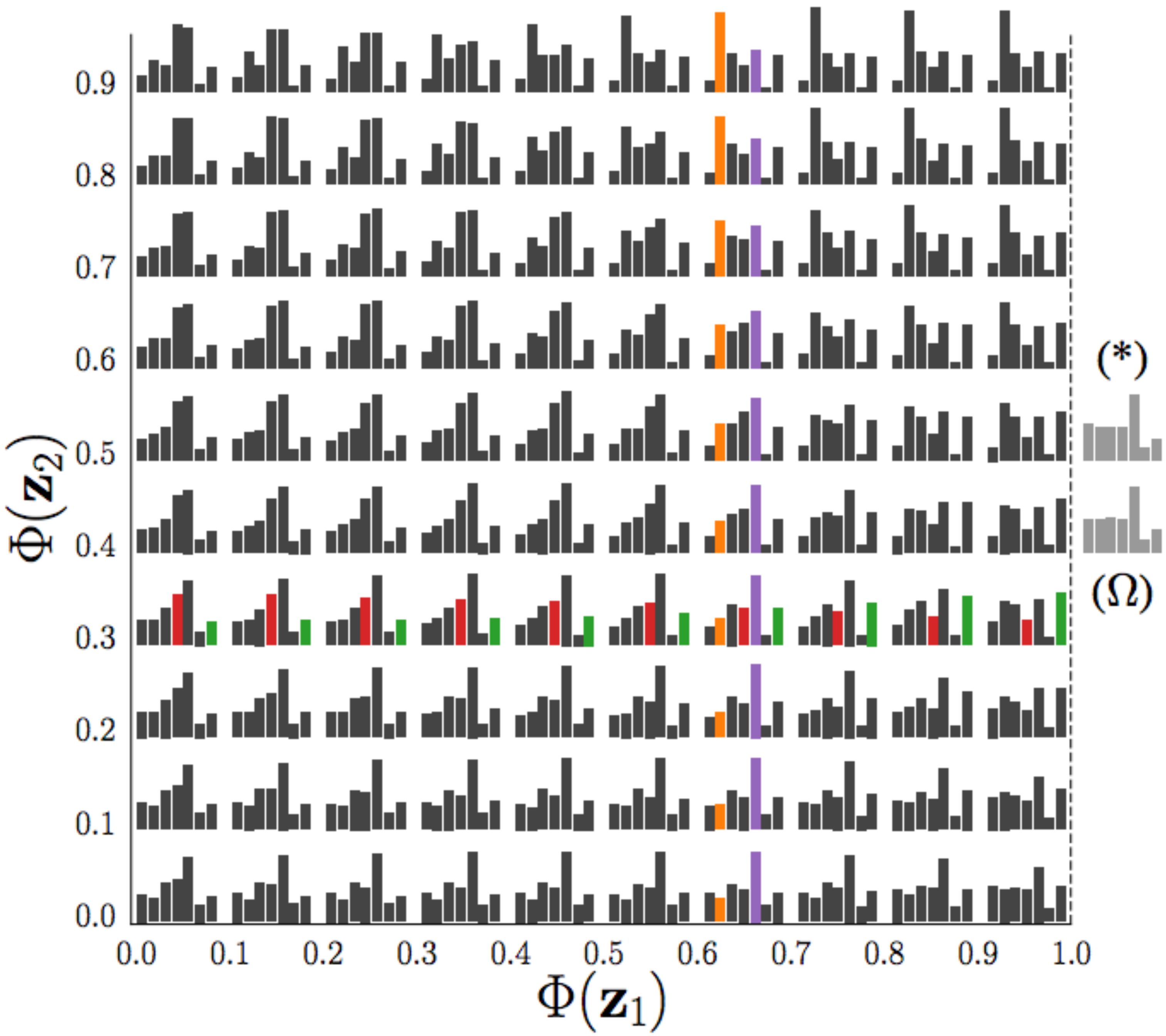
(a) Avg. degree of start node

(b) Avg. share of nodes in start community

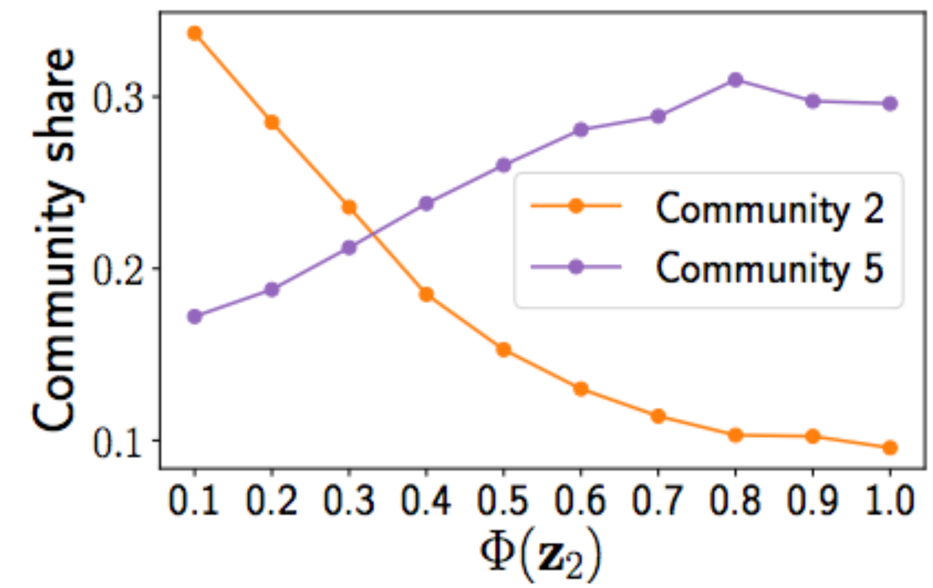
(c) Gini coefficient (input graph: 0.48)

(d) Max. degree (input graph: 240)

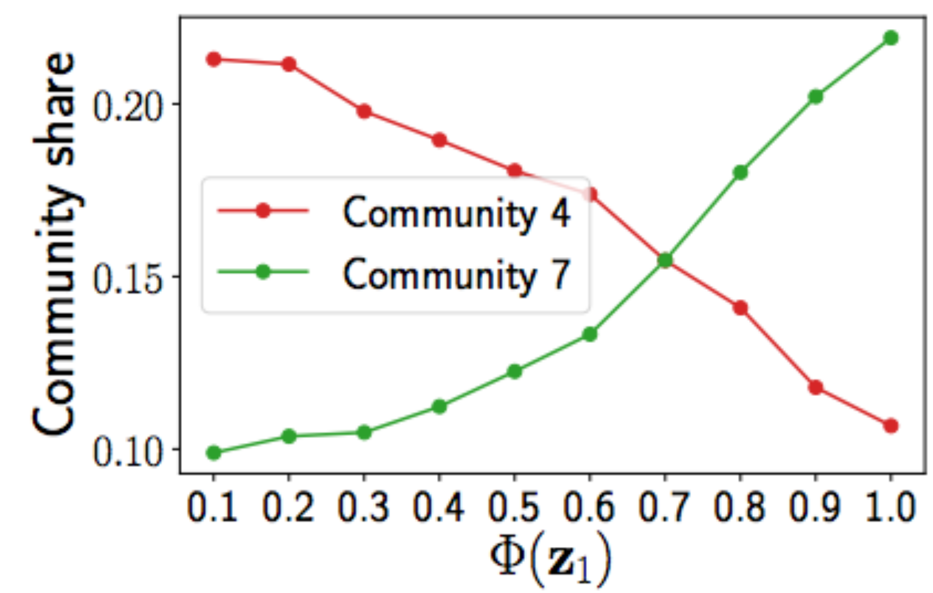
Further Analysis



(a) Community histograms



(b) Top to bottom trajectory



(c) Left to right trajectory

Discussion

- The NetGAN approach is based exclusively on the random-walk approach for graph node representation introduced by node2vec
- Generated graphs are on par/outperform previous explicit methods
- This approach is limited to subgraph generation and it will probably perform on par or worse then SMILES based approaches for molecule generation