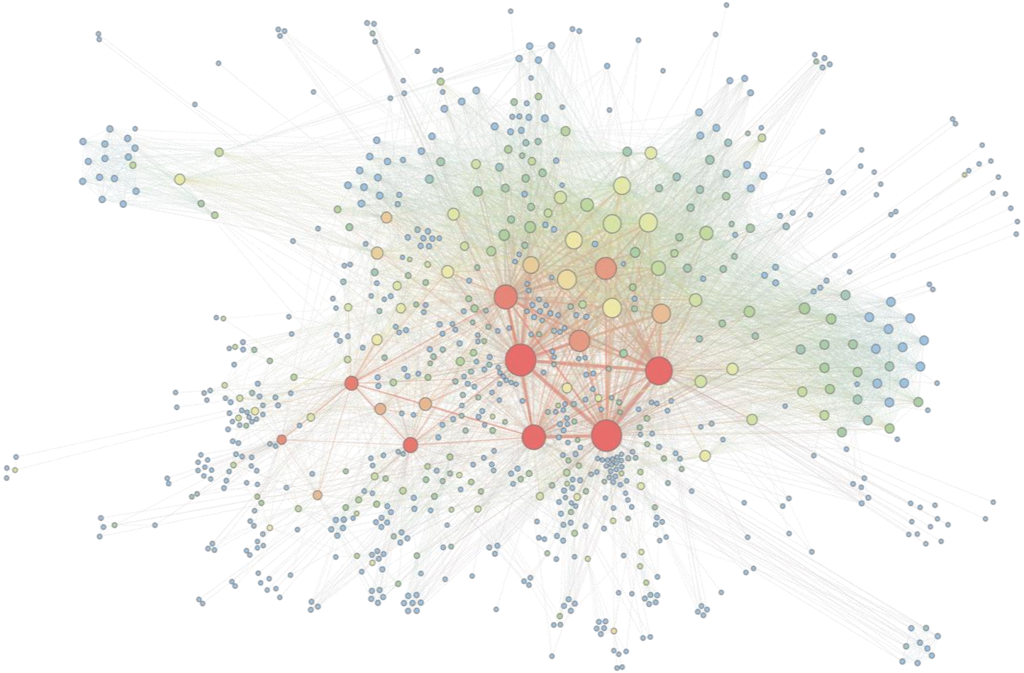
Exploiting Node Content for Multiview Graph Convolutional Network and Adversarial Regularization

Qiuhao Lu[1], Nisansa de Silva[1], Dejing Dou[1], Thien Huu Nguyen[1],
Prithviraj Sen[2], Berthold Reinwald[2], Yunyao Li[2]
[1]Department of Computer and Information Science, University of Oregon
[2]IBM Research - Almaden

Introduction

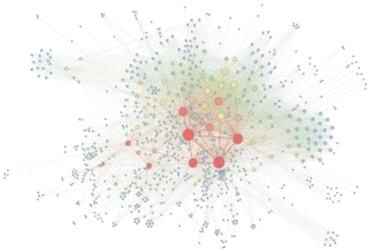
- Networks are everywhere
  - citation networks
  - social networks
  - drug-target-interaction networks
  - mobile networks

- Tasks
  - link prediction
  - node clustering
  - many other downstream tasks



Introduction

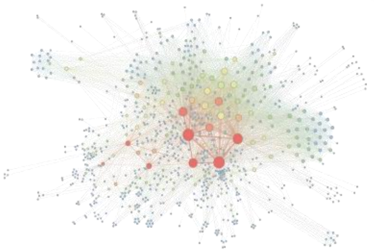
- (semi-) Supervised vs. Unsupervised



Representations

Task

Vs.



Representations

Task1

Task2

Task3

Network Representation Learning

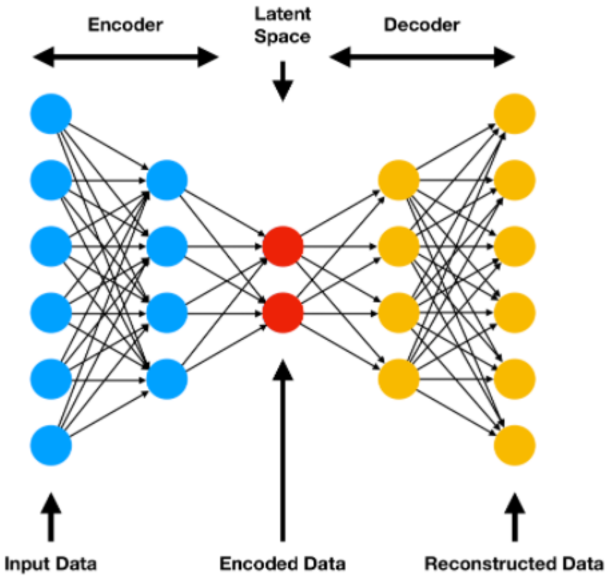
- A unified encoder – decoder framework [1]
  - An encoder maps each node to a low-dimensional vector
    - $\mathrm{ENC}\colon\ \mathcal{V} \to \mathbb{R}^d$
  - A decoder decodes information about the graph from the learned embeddings. (pairwise decoder mostly)
    - $\mathrm{DEC}\colon \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$
  - Optimization
    - $\mathcal{L} = \sum_{(v_i,v_j)\in\mathcal{D}} \ell\left(\mathrm{DEC}\left(\boldsymbol{z_i}, \boldsymbol{z_j}\right), s_{\mathcal{G}}(v_i, v_j)\right)$

[1] Hamilton, William L., Rex Ying, and Jure Leskovec. "Representation learning on graphs: Methods and applications." *arXiv preprint arXiv:1709.05584* (2017).

Network Representation Learning

- Shallow models
  - Encoder function is a simple “lookup”:
    - $\mathrm{ENC}(v_i) = \mathbf{Z}\,\boldsymbol{v}_i$
  - $\mathbf{Z}$ is optimized directly, i.e., the trainable parameters
  - Matrix-factorization and Random walk approaches

- Deep models - Autoencoders
  - Parameter-sharing encoders, e.g., DNN, GCN, etc.



- They do not explicitly consider the *semantic relatedness* between nodes

Image from https://www.compthree.com/blog/autoencoder.

Multiview Adversarially Regularized Graph Autoencoder (MRGAE)

- Preliminaries
  - Graph Convolutional Network (GCN)
    - $\mathbf{H}^{(l+1)} = f\left(\mathbf{H}^{(l)}, \mathbf{A}\right)$
      $= \sigma(\widehat{\mathbf{D}}^{-\frac{1}{2}}\widehat{\mathbf{A}}\widehat{\mathbf{D}}^{\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)})$
  - Adversarial Regularization
    - Encoder as generator
    - Incorporate a MLP as discriminator
    - Distribution consistent

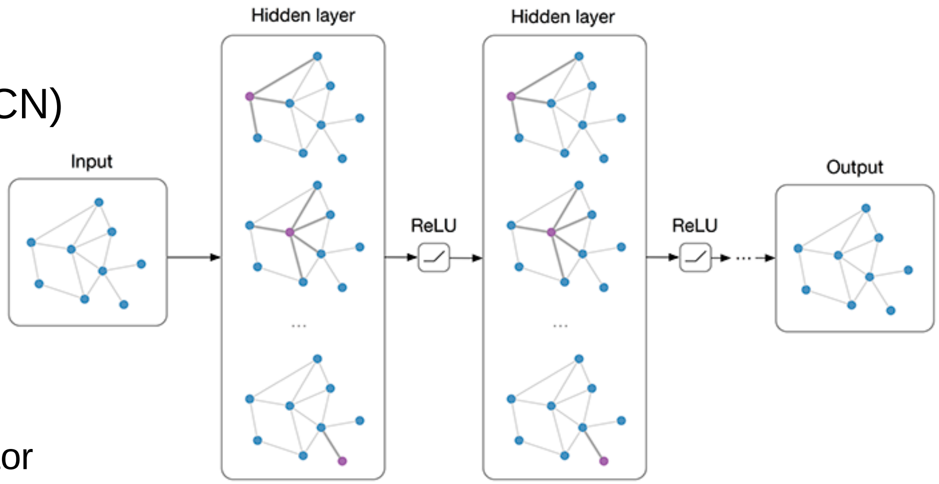


Image from https://tkipf.github.io/graph-convolutional-networks.

Multiview Adversarially Regularized Graph Autoencoder (MRGAE)

- Second View Construction
  - cosine similarity
- Encoder – Decoder
  - Multiview reconstruction loss (MRL)
- Multiview Adversarial Regularization
  - Normal Discriminator
  - View Discriminator
- Training objective
  - reconstruction loss
  - adversarial loss

$$L = L_{mrl} + \gamma L_{adv} + L_{reg}$$

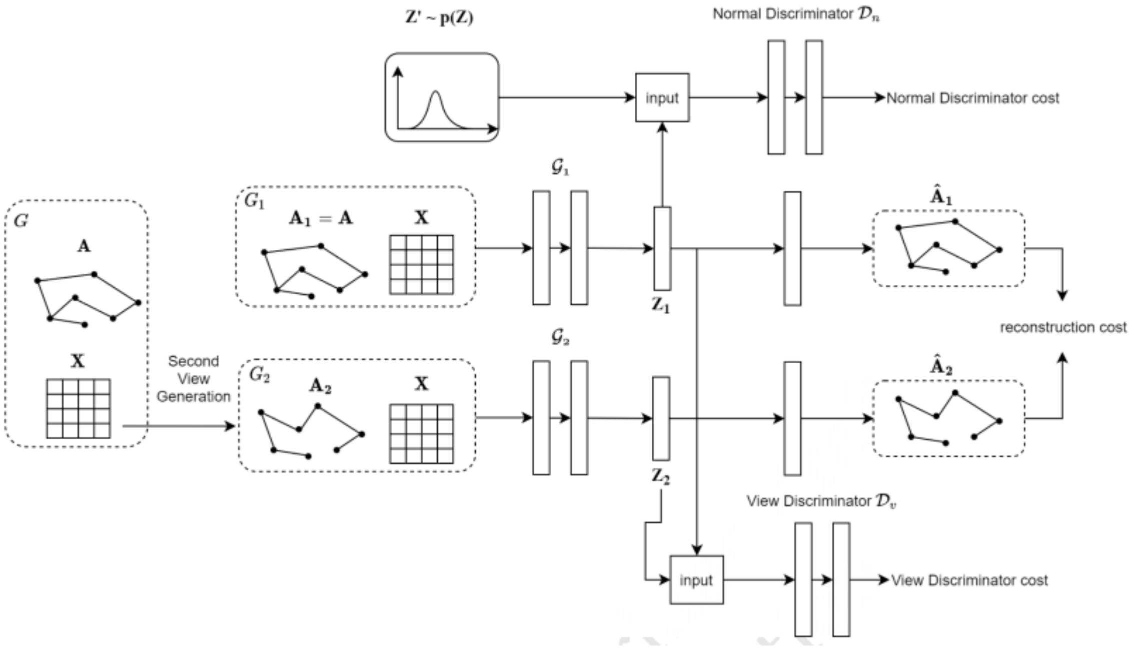


Figure 1: Architecture of MRGAE.

Evaluation: Link Prediction

- Link prediction
  - Benchmark dataset
    - Scientific publication citation networks
    - Each node is described by BOW representations
    - Each node has a corresponding class label
  - Experiment settings
    - 32-dim hidden layer, 16-dim embedding layer for the GCN encoders
    - Multi-layer perceptron (MLP) with two 128-dim hidden layers for the two discriminators

Table 1: Citation networks.

Dataset	Nodes	Edges	Features	Classes
Cora	2,708	5,429	1,433	7
Citeseer	3,327	4,732	3,703	6
Pubmed	19,717	44,338	500	3

Evaluation: Link Prediction

Table 2: Performance comparison on link prediction.

Method	Cora		Citeseer		Pubmed	
	AUC	AP	AUC	AP	AUC	AP
SC	84.6 ± 0.01	88.5 ± 0.00	80.5 ± 0.01	85.0 ± 0.01	84.2 ± 0.02	87.8 ± 0.01
DW	83.1 ± 0.01	85.0 ± 0.00	80.5 ± 0.02	83.6 ± 0.01	84.2 ± 0.00	84.1 ± 0.00
GAE	91.0 ± 0.02	92.0 ± 0.03	89.5 ± 0.04	89.9 ± 0.05	96.4 ± 0.00	96.5 ± 0.00
VGAE	91.4 ± 0.01	92.6 ± 0.01	90.8 ± 0.02	92.0 ± 0.02	94.4 ± 0.02	94.7 ± 0.02
ARGA	92.4 ± 0.003	93.2 ± 0.003	91.9 ± 0.003	93.0 ± 0.003	96.8 ± 0.001	97.1 ± 0.001
ARVGA	92.4 ± 0.004	92.6 ± 0.004	92.4 ± 0.003	93.0 ± 0.003	96.5 ± 0.001	96.8 ± 0.001
MRGAE	94.0 ± 0.7	94.1 ± 0.6	94.3 ± 0.4	94.9 ± 0.8	97.2 ± 0.2	97.4 ± 0.3
DBGAN	94.5 ± 0.01	95.1 ± 0.05	94.5 ± 0.04	95.8 ± 0.01	96.8 ± 0.01	97.3 ± 0.02
MRGAE*	**95.0 ± 0.3**	**95.2 ± 0.4**	**95.7 ± 0.5**	**96.4 ± 0.4**	**97.8 ± 0.1**	**97.8 ± 0.2**

Evaluation: Node Clustering

- Node Clustering
  - Benchmark dataset
    - Scientific publication citation networks
    - Each node is described by BOW representations
    - Each node has a corresponding class label
  - Experiment settings
    - 32-dim hidden layer, 16-dim embedding layer for the GCN encoders
    - Multi-layer perceptron (MLP) with two 128-dim hidden layers for the two discriminators

Table 1: Citation networks.

Dataset	Nodes	Edges	Features	Classes
Cora	2,708	5,429	1,433	7
Citeseer	3,327	4,732	3,703	6
Pubmed	19,717	44,338	500	3

Evaluation: Node Clustering

Table 3: Node clustering performance on Cora.

Method	Acc	NMI	F1	Prec	ARI
SC	0.367	0.127	0.318	0.193	0.031
DW	0.484	0.327	0.392	0.361	0.243
RTM	0.440	0.230	0.307	0.332	0.169
RMSC	0.407	0.255	0.331	0.227	0.090
TADW	0.560	0.441	0.481	0.396	0.332
GAE	0.596	0.429	0.595	0.596	0.347
VGAE	0.609	0.436	0.609	0.609	0.346
ARGA	0.640	0.449	0.619	0.646	0.352
ARVGA	0.638	0.450	0.627	0.624	0.374
MRGAE	0.703	0.523	0.681	0.716	0.476
GALA	0.745	**0.576**	–	–	0.531
DBGAN	0.748	0.560	–	–	0.540
MRGAE*	**0.764**	0.559	**0.740**	**0.742**	**0.570**

Table 4: Node clustering performance on Citeseer.

Method	Acc	NMI	F1	Prec	ARI
SC	0.239	0.056	0.299	0.179	0.010
DW	0.337	0.088	0.270	0.248	0.092
RTM	0.451	0.239	0.342	0.349	0.203
RMSC	0.295	0.139	0.320	0.204	0.049
TADW	0.455	0.291	0.414	0.312	0.228
GAE	0.408	0.176	0.372	0.418	0.124
VGAE	0.344	0.156	0.308	0.349	0.093
ARGA	0.573	0.350	0.546	0.573	0.341
ARVGA	0.544	0.261	0.529	0.549	0.245
MRGAE	0.627	0.361	0.587	0.601	0.363
GALA	**0.693**	**0.441**	–	–	**0.446**
DBGAN	0.670	0.407	–	–	0.414
MRGAE*	0.671	0.403	**0.620**	**0.620**	0.418

Evaluation: Readmission Prediction

- Lin et al.'s work [2] gives the state-of-the-art performance in unplanned 30-day all-cause ICU readmission prediction on MIMIC-III dataset.
  - MIMIC-III Dataset
    - a large, freely-available database comprising deidentified health-related data associated with over 40,000 ICU patients, which includes information such as demographics, vital sign measurements made at the bedside (~1 data point per hour), laboratory test results, procedures, medications, caregiver notes, imaging reports, and mortality
  - Method
    - Features used include 17 types of selected chart events (diastolic blood pressure, capillary refill rate, etc.) over a 48-hour time window, **ICD-9 embeddings of diagnosis**, and demographic information of patients
    - Input features are fed into a LSTM-CNN model as shown in Figure 3
  - Metric
    - The Area under the Receiver Operating Characteristic (AUROC) as primary metric. It measures the overall performance of the recall with respect to different false positive rate.
    - AUROC is a commonly used metric for the readmission prediction task.
    - Along with some other metrics, including positive case recall rate (sensitivity), accuracy, etc.

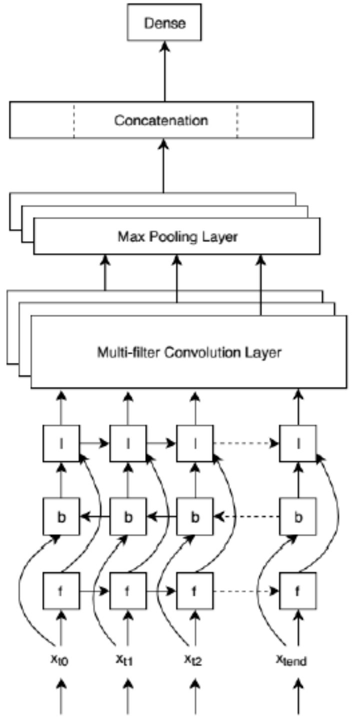


Figure 2: Lin et al.'s LSTM-CNN model

[2] Lin, Yu-Wei, et al. "Analysis and prediction of unplanned intensive care unit readmission using recurrent neural networks with long short-term memory." *PloS one* 14.7 (2019): e0218942.

Evaluation: Readmission Prediction

- ICD-9 is the 9-th revision of the International Statistical Classification of Diseases and Related Health Problems (ICD), a medical classification list by WHO.
  - Nodes represent diagnosis (disease) and are described by short textual descriptions.
  - Edges represent "is-a" relationships.
- Second View Construction
  - Transform the short textual descriptions of nodes into one-hot representations
  - If the cosine similarity between two nodes is greater than 0.7, we create a link between them in the second view of ICD-9

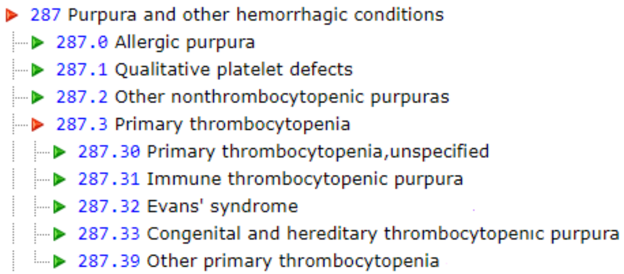


Figure 3: Part of ICD-9.

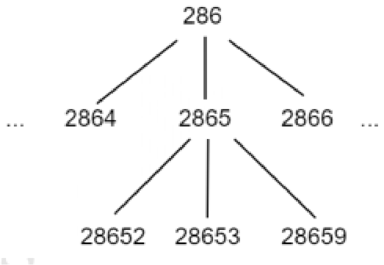


Figure 4: Part of ICD-9 tree structure.

Research Methods: Readmission Prediction

Table 5: Unplanned 30-day Readmission Prediction Performance with Different Embeddings

Method	Acc	Pre-0	Pre-1	Re-0	**Re-1**	**A.R**	A.P
Poincaré	0.7223	0.9035	0.3740	0.7361	0.6655	0.7786	0.4827
GAE	0.7052	0.9061	0.3597	0.7089	0.6901	0.7712	0.4588
VGAE	0.7042	0.9007	0.3554	0.7127	0.6684	0.7653	0.4444
ARGA	0.7075	0.9126	0.3654	0.7057	0.7150	0.7757	0.4593
ARVGA	0.6966	0.9056	0.3518	0.6973	0.6934	0.7693	0.4519
MRGAE	0.7094	0.9157	0.3687	0.7055	**0.7259**	**0.7807**	0.4770

*Acc: Accuracy, Pre: Precision, Re: Recall, A.R: AUC under ROC, A.P: AUC under PRC

Ablation Study

- Ablated methods
  - View Discriminator removed
  - MRL removed
  - Both removed

Table 6: Ablation analysis.

Method	Link Prediction		Node Clustering				
	AUC	AP	Acc	NMI	F1	Prec	ARI
w/o $\mathcal{D}_v$	93.8	94.4	0.748	0.540	0.732	0.744	0.526
w/o MRL	94.0	94.1	0.706	0.527	0.686	0.712	0.496
w/o both	92.9	93.2	0.643	0.482	0.645	0.664	0.397
MRGAE	94.4	94.7	0.764	0.559	0.740	0.742	0.570

Future Work

- The proposed network embedding method, i.e., MRGAE, can be extended to a relational version which applies to graphs with multiple types of edges.

- It is interesting to see if any performance gain can be achieved if we add a regularization term to existing embedding methods that explicitly preserves semantic relatedness between nodes.

Thank you for your attention!