GNE: A deep framework for gene network inference by aggregating biological information

Kishan K C (kk3671@rit.edu)[1] Rui Li[1] Feng Cui[2] Anne R. Haake[1]

[1]Golisano College of Computing and Information Sciences [2]Thomas H. Gosnell School of Life Sciences

R·I·T
B. Thomas Golisano College of COMPUTING AND INFORMATION SCIENCES

Abstract

- Understanding **functional aspects of genes or proteins** is crucial to provide insights into underlying biological phenomenon for different health and disease conditions.
- **Often intractable** through biological experiments.
- We propose **Gene Network Embedding (GNE)**, a deep neural network architecture to learn lower dimensional representation for each gene, by **integrating the topological properties** of gene interaction network with additional information such as **expression data**.
- This approach models **complex statistical relationships** between interaction data and gene expression, which addresses the problem of sparsity in interaction data.
- Experimental results show that the model learns a **comprehensive** representation for gene that improves the performance in genetic interaction prediction for yeast and E. coli datasets.
- Moreover, a set of novel gene interactions predictions are validated by up-to-date literature-based database entries.

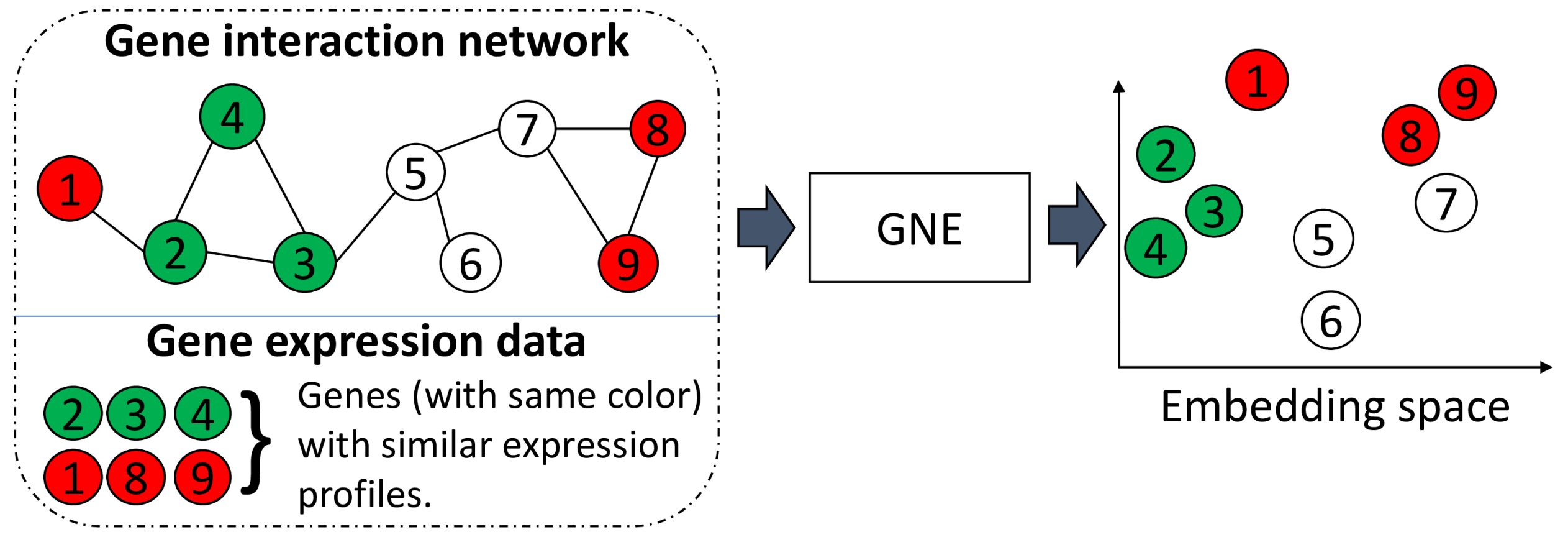
Datasets

Gene interaction data from BioGRID interaction database and gene expression data from DREAM5 challenge.

	Interaction Network Data		Expression data
Organism	# Genes	# Interactions	# Experiments
Yeast	5,950	544,652	536
E. coli	4,511	148,340	805

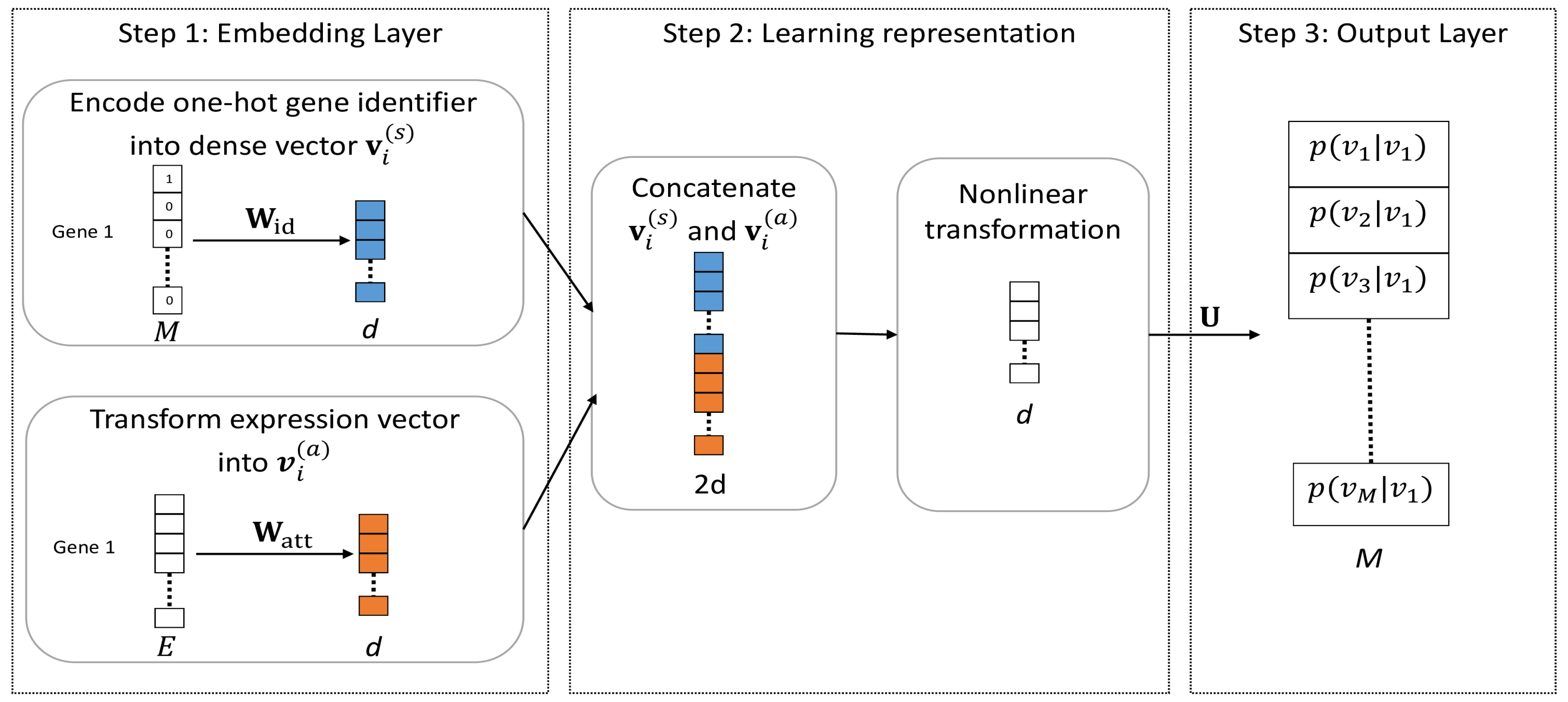
GNE Overview

Given a gene network denoted as $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{A})$, gene network embedding aims to learn a function $\boldsymbol{f}$ that maps gene network structure and their attribute information to a $\boldsymbol{d}$-dimensional space where a gene is represented by a vector $\boldsymbol{y}_i \,\epsilon\, \mathbb{R}^d$ where $\boldsymbol{d} \ll \boldsymbol{M}$. The low dimensional vectors $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ for genes $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$ preserve their relationships in terms of the network topological structure and attribute proximity.



An illustration of Gene Network Embedding (GNE) to integrate gene interaction network and gene expression data to learn lower dimensional representation of gene.

GNE Architecture


Step 1: Embedding Layer
Encode one-hot gene identifier into dense vector $\mathbf{v}_i^{(s)}$
Gene 1
1
0
0
0
$\mathbf{W}_{\mathrm{id}}$
M
d
Transform expression vector into $\boldsymbol{v}_i^{(a)}$
Gene 1
$\mathbf{W}_{\mathrm{att}}$
E
d
Step 2: Learning representation
Concatenate $\mathbf{v}_i^{(s)}$ and $\mathbf{v}_i^{(a)}$
2d
Nonlinear transformation
d
$\mathbf{U}$
Step 3: Output Layer
$p(v_1|v_1)$
$p(v_2|v_1)$
$p(v_3|v_1)$
$p(v_M|v_1)$
M

Overview of Gene Network Embedding (GNE) Framework for gene interaction prediction.

Step 1: Embedding Layer

- **GNE Network Structure Modeling**

We encode one-hot encoded representation of a gene v_i via embedding lookup.

$$\mathbf{v}_i^{(s)} = \mathbf{W}_{id} v_i$$

- **GNE Expression Modeling**

We use Exponential Linear unit (ELU) to model non-linearity of gene expression x_i and capture underlying patterns.

$$\mathbf{v}_i^{(a)} = \mathrm{elu}(\mathbf{W}_{att} \cdot x_i)$$

Step 2: GNE Integration

- Concatenation of structural and attribute representation

$$\mathbf{v}_i = [\mathbf{v}_i^{(s)} \quad \lambda \mathbf{v}_i^{(a)}]$$

- Transformation of concatenated representation via $\boldsymbol{k}$-hidden layers with hyperbolic tangent activation.

$$\mathbf{h}_i^{(k)} = \delta_k(\mathbf{W}_k \mathbf{h}_i^{(k-1)} + b^{(k-1)})$$

Step 3: Output layer

- Last layer outputs the probability vector which contains conditional probability of all other genes to gene v_i

$$\mathbf{o}_i = [p(v_1|v_i), p(v_2|v_i), \ldots, p(v_M|v_i)]$$

where

$$p(v_j|v_i) = \frac{\exp(\widetilde{\mathbf{v}}_j \cdot \mathbf{h}_i^{(k)})}{\sum_{j'=1}^{M} \exp(\widetilde{\mathbf{v}}_{j'} \cdot \mathbf{h}_i^{(k)})}$$

Optimization

$$\Theta^* = \underset{\Theta}{\mathrm{argmax}} \left[\sum_{i=1}^{M} \sum_{v_j \,\in\, N_i} \log \frac{\exp(\widetilde{\mathbf{v}}_j \cdot \mathbf{h}_i^{(k)})}{\sum_{j'=1}^{M} \exp(\widetilde{\mathbf{v}}_{j'} \cdot \mathbf{h}_i^{(k)})} \right]$$

Acknowledgements

This material is based upon work supported by National Science under Grant NSF-1062422.

Results

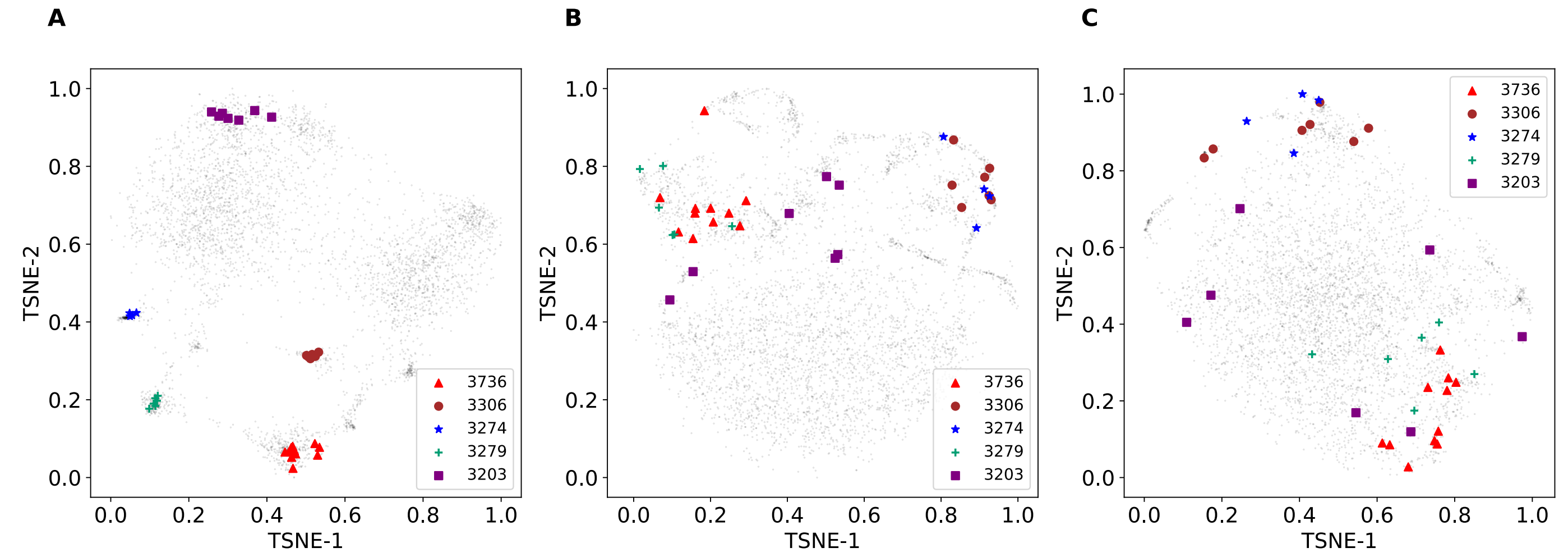
AUROC comparison shows that GNE outperforms other strong baselines.

Methods	Yeast		E. coli	
	AUROC	AUPR	AUROC	AUPR
Isomap	0.507	0.588	0.559	0.672
LINE	0.726	0.686	0.897	0.851
node2vec	0.739	0.708	0.912	0.862
GNE*	0.787	0.784	0.930	0.931
GNE	**0.825**	**0.821**	**0.940**	**0.939**

Temporal holdout validation

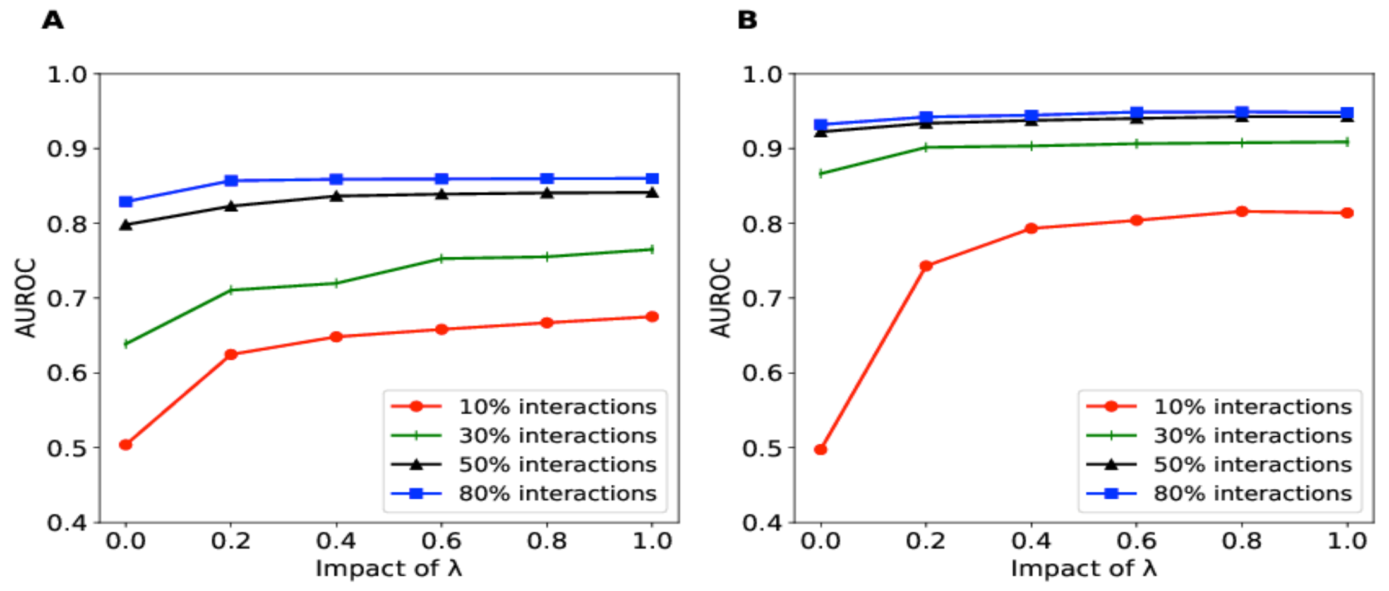
Methods	Yeast		E. coli	
	AUROC	AUPR	AUROC	AUPR
LINE	0.620	0.611	0.569	0.598
node2vec	0.640	0.609	0.587	0.599
GNE	**0.710**	**0.683**	**0.653**	**0.658**

Visualization of learned representation



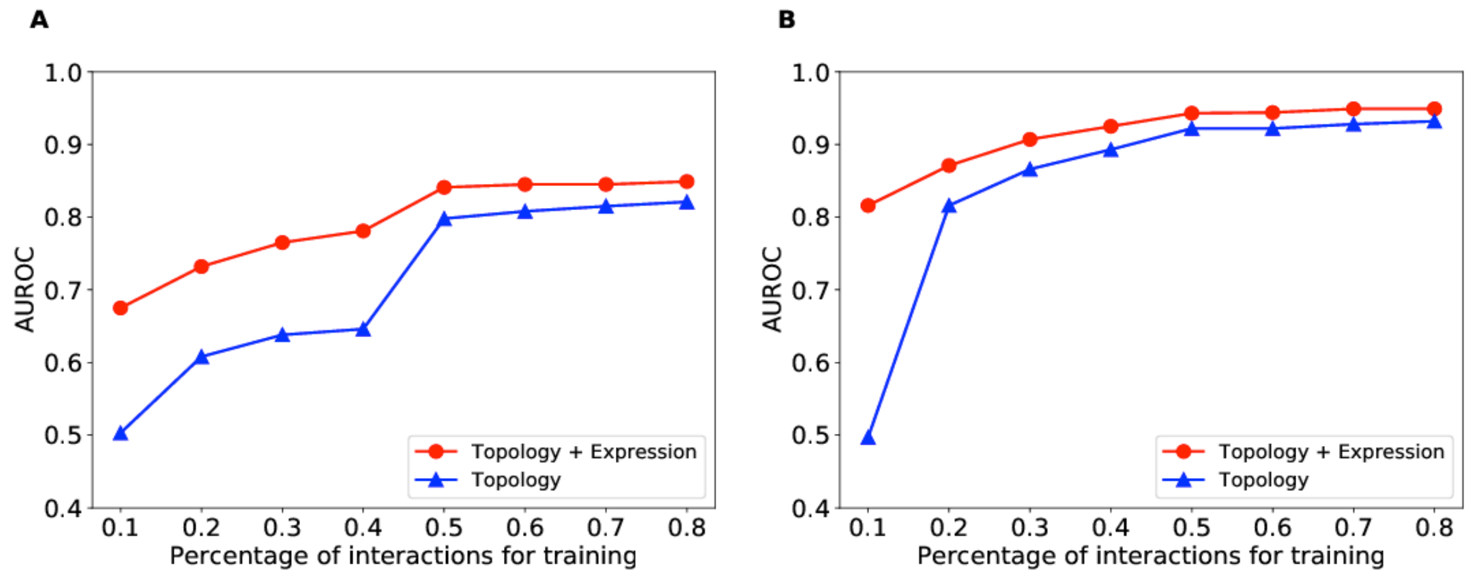
*Visualization of learned embeddings for genes on E. coli using (**A**) GNE, (**B**) LINE, and (**C**) node2vec.*

Impact of λ (Relative importance of topology and expression data)



*Impact of λ on performance of our method trained with different percentages of interactions for training GNE. (**A**) yeast (**B**) E. coli.*

Impact of network sparsity



*AUROC comparison of our method's performance with respect to network sparsity and integration of expression data. (**A**) yeast (**B**) E. coli.*