Chemical Property Prediction via Graph Knowledge Transfer

Zhenbang Wu[12*], Haonan Wang[2*], Ziniu Hu[3], Yizhou Sun[3]

[1] Zhejiang University

[2] University of Illinois at Urbana-Champaign

[3] University of California, Los Angeles

* Equal Contribution

{zw12, haonan3}@illinois.edu, yzsun@cs.ucla.edu

CONTENT

- Introduction
- Related Work
  - Graph-Level Classification / Regression
  - Transfer Learning
  - Multi-Task Learning
- Method
  - Hypergraph Knowledge Transfer
- Experiment
- Next Step
- Conclusion

Introduction

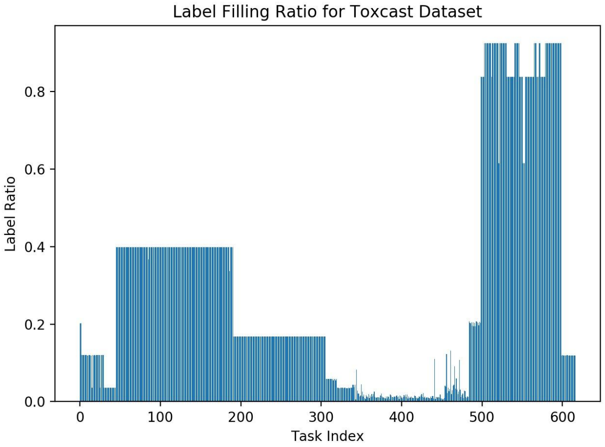
Introduction

- Predicting molecular properties
  - A fundamental problem in Biomedicine and Chemistry
  - Expensive and time-consuming
- Use of deep learning (DL)
  - Speed-up the process
  - Better predict molecular properties

Mengying Sun, Sendong Zhao, Coryandar Gilvary, Olivier Elemento, Jiayu Zhou, Fei Wang, Graph convolutional networks for computational drug development and discovery, Briefings in Bioinformatics, , bbz042, https://doi.org/10.1093/bib/bbz042

Introduction

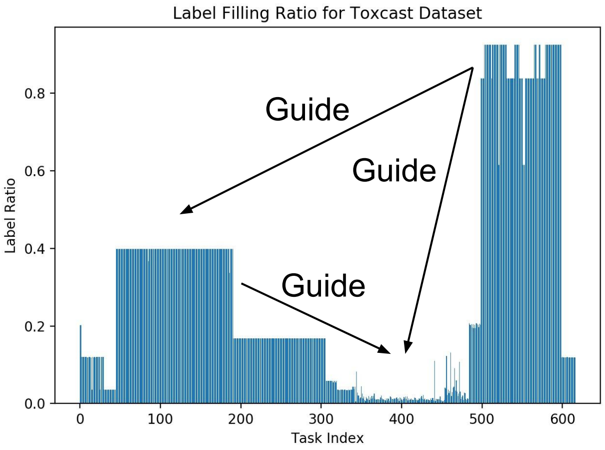
- Practical effect of DL is limited
  - Require large amounts of labeled data
- Usually, in Biomedicine and Chemistry
  - Fully labeling a dataset is unaffordable [1]
  - Label ratio between properties is imbalanced [2]

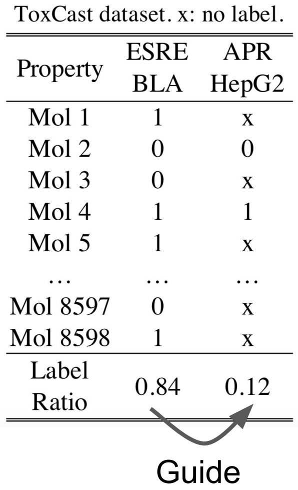


[1] H. Altae-Tran, B. Ramsundar, A. S. Pappu, V. Pande. Low Data Drug Discovery with One-Shot Learning. ACS Cent. Sci. 2017, 3 (4), 283−293.
[2] W. Lin, D. Xu, Imbalanced multi-label learning for identifying antimicrobial peptides and their functional types. Bioinformatics, Volume 32, Issue 24, 15 December 2016, Pages 3745.

Introduction

- Intuition: leverage task dependency
  - ○ Knowledge extracted from fully labeled properties can enhance the prediction of properties with few labels



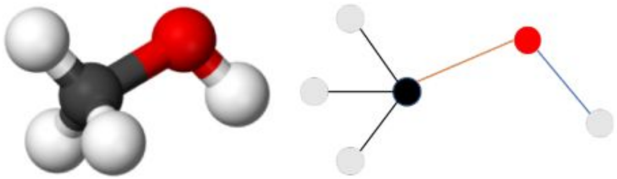

ToxCast dataset. x: no label.

Property	ESRE BLA	APR HepG2
Mol 1	1	x
Mol 2	0	0
Mol 3	0	x
Mol 4	1	1
Mol 5	1	x
…	…	…
Mol 8597	0	x
Mol 8598	1	x
Label Ratio	0.84	0.12

Related Work:
Graph-Level Classification / Regression

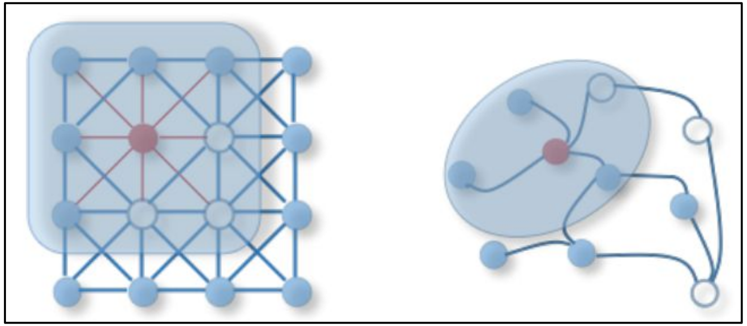
Graph-Level Classification / Regression

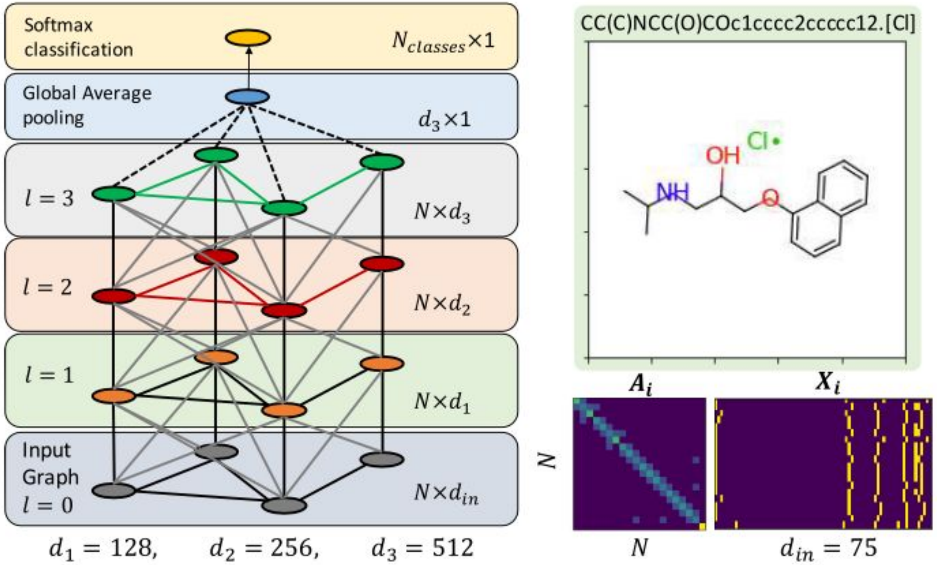
- Molecule graph
  - Atoms -> nodes
  - Chemical bonds -> edges



- Molecule -> Graph-Level Prediction
  - e.g. toxicity, solubility, side effect

Graph Classification / Regression Model





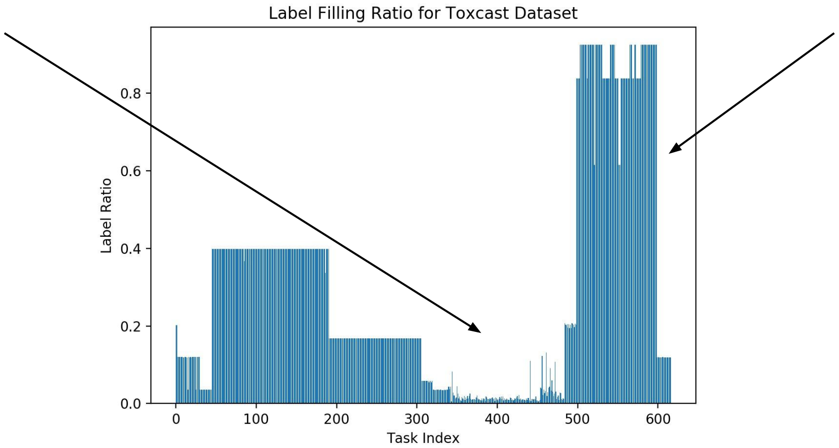
Suffer from the lack of labeled data

Z. Wu *et al.*, “A Comprehensive Survey on Graph Neural Networks.”
Pope, P.E.; Kolouri, S.; Rostami, M.; Martin, C.E.; Hoffmann, H. Explainability Methods for Graph Convolutional Neural Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019.

Related Work: Transfer Learning

Transfer Learning

- Pre-train the model on properties with abundant labels
- Fine-tune the model on properties with few labels



Fine-Tune

Pre-Train

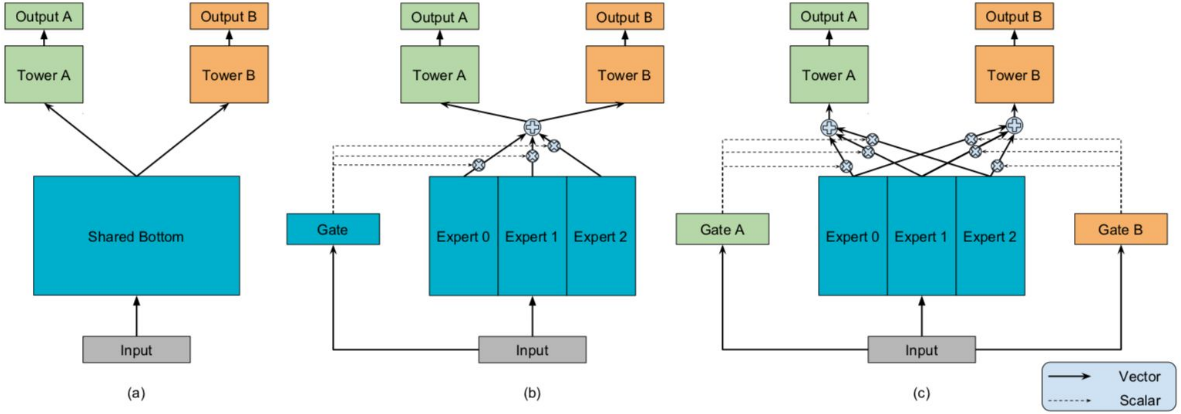
Not End-to-End

Not Time-Efficient

Related Work: Multi-Task Learning

Multi-Task Learning

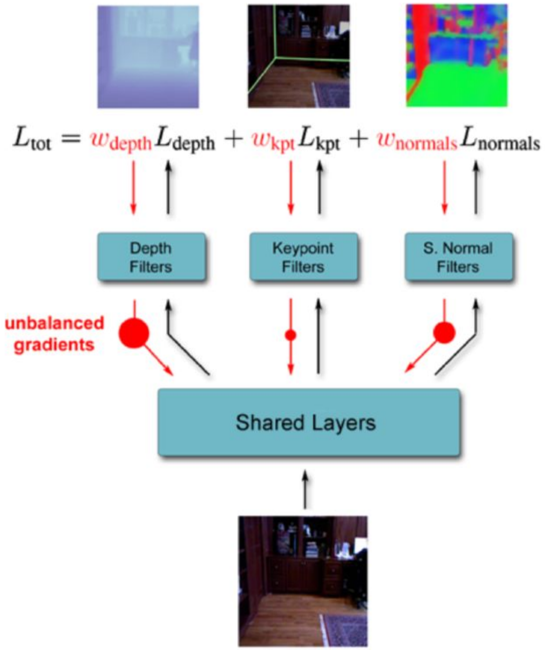
- Train multiple tasks together
  - Time efficient
  - Leverate knowledge among tasks



Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '18). ACM, New York, NY, USA, 1930-1939. DOI: https://doi.org/10.1145/3219819.3220007

Multi-Task Learning

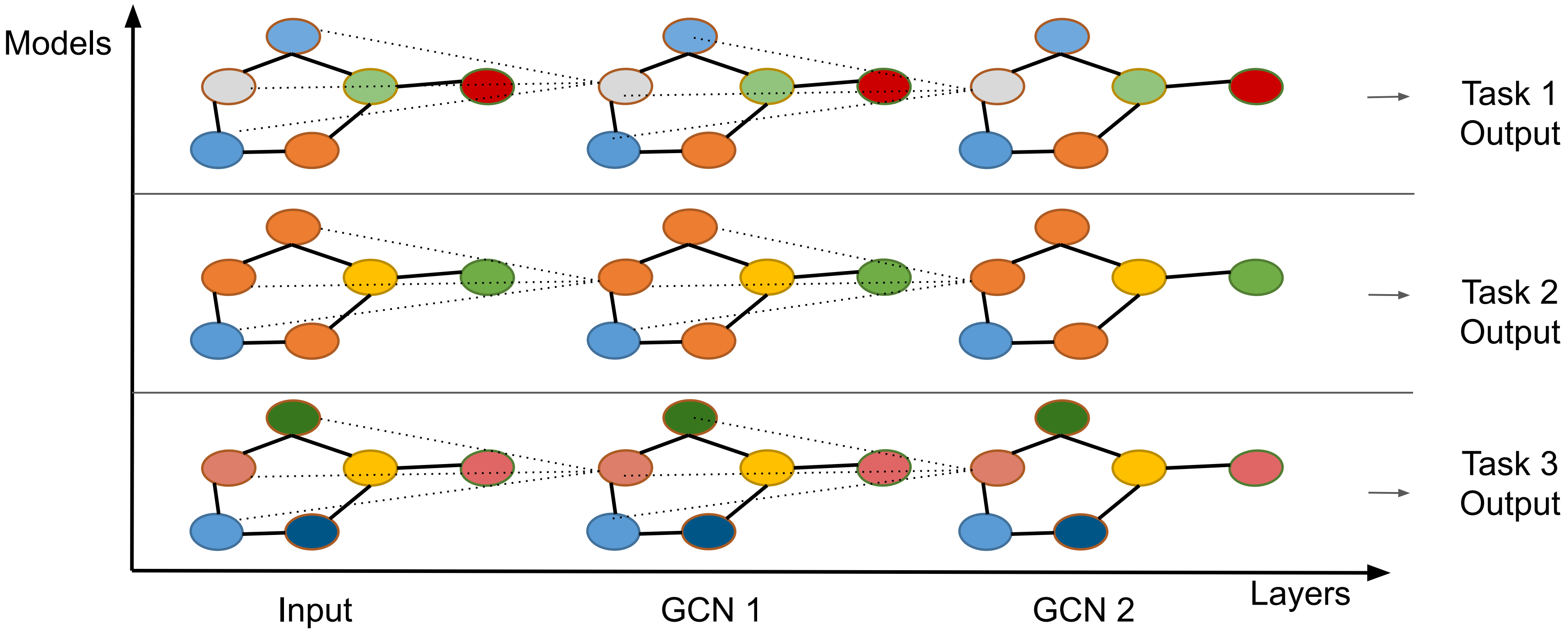
- Imbalanced label ratio
- Imbalanced gradients
  - Tasks with abundant labels / larger gradients will dominate the model
- Neglect interaction among different tasks


$L_{\text{tot}} = w_{\text{depth}} L_{\text{depth}} + w_{\text{kpt}} L_{\text{kpt}} + w_{\text{normals}} L_{\text{normals}}$
Depth Filters
Keypoint Filters
S. Normal Filters
unbalanced gradients
Shared Layers

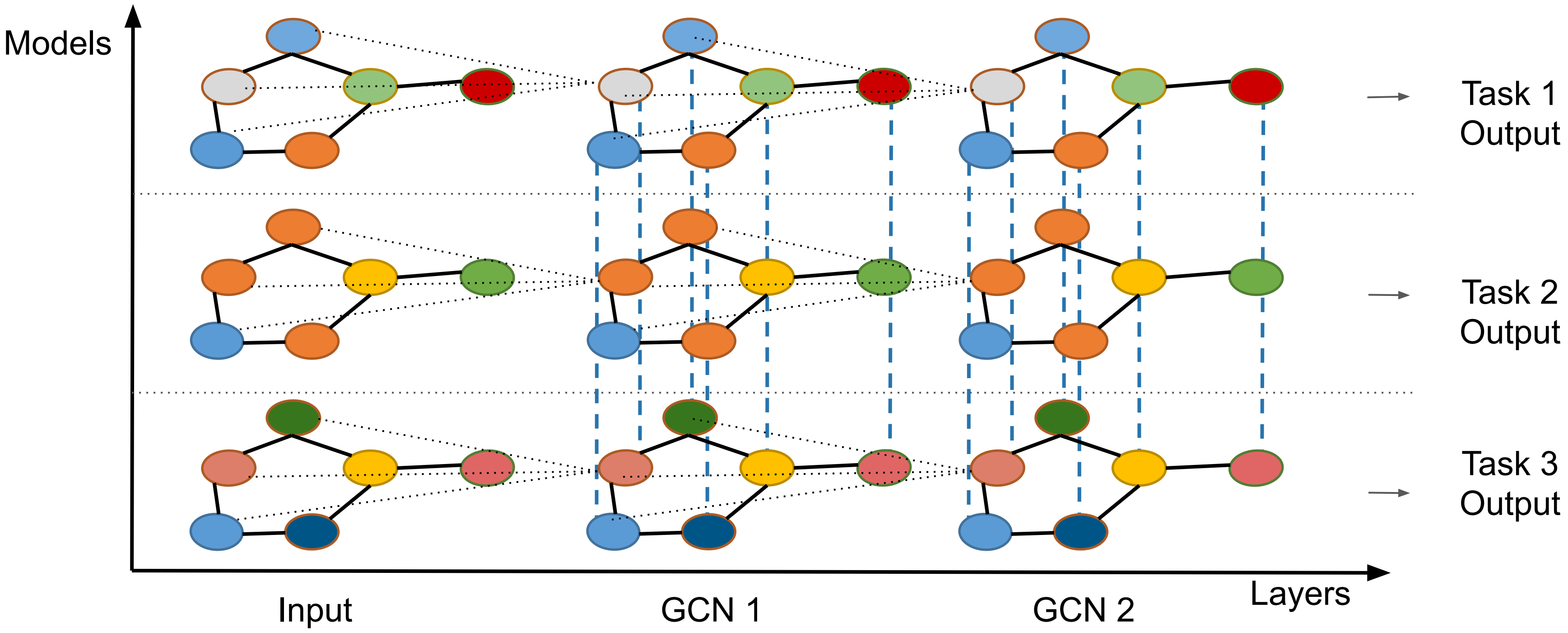
Zhao Chen Vijay Badrinarayanan Chen-Yu Lee Andrew Rabinovich GradNorm: Gradient Normalization for Adaptive Loss Balancing in Deep Multitask Networks. 2017 abs/1711.02257 CoRR
http://arxiv.org/abs/1711.02257 db/journals/corr/corr1711.html#abs-1711-02257

Method: Hypergraph Knowledge Transfer

Normal GCN


Models
Task 1 Output
Task 2 Output
Task 3 Output
Input
GCN 1
GCN 2
Layers

Hypergraph GCN



Data-Dependent Task Dependency Graph

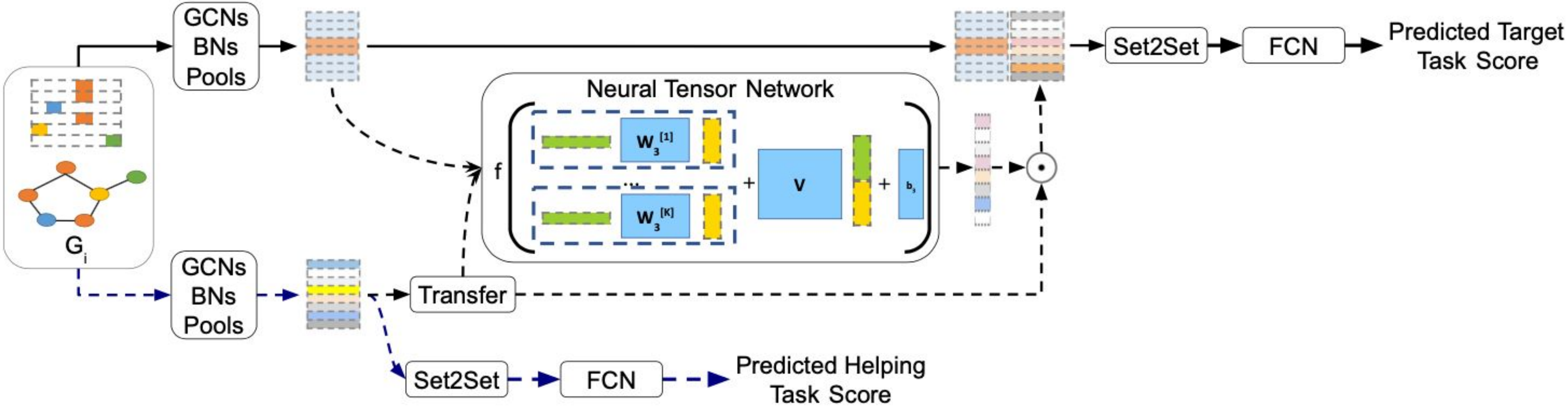
Hypergraph Knowledge Transfer

- Each task will have its own base model
- Calculate the data-dependent task dependency graph
- Aggregate representation from different task specific models

↓

- Mutually enhance performance on all tasks
- Mining the task-level dependency

Base Model (Dual-Task)



Stage I
Node
Embed.

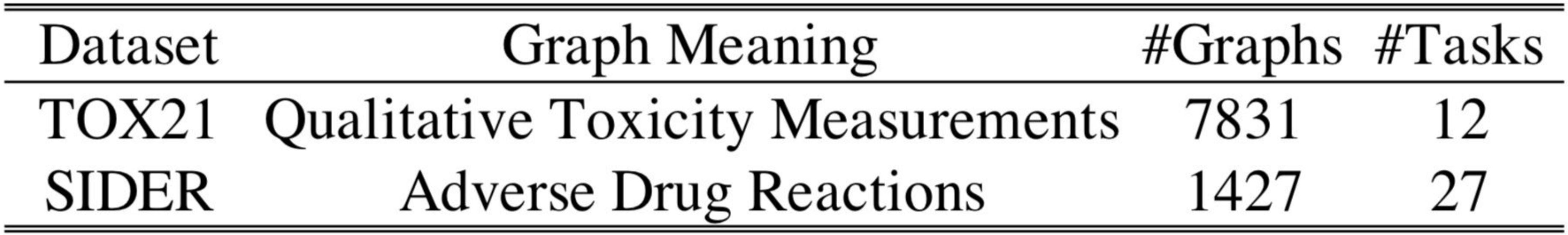
Stage II
Knowledge
Transfer

Stage III
Knowledge
Fusion

Stage IV
Graph
Embed.

Experiment

Experiment Setting



Dataset	Graph Meaning	#Graphs	#Tasks
TOX21	Qualitative Toxicity Measurements	7831	12
SIDER	Adverse Drug Reactions	1427	27

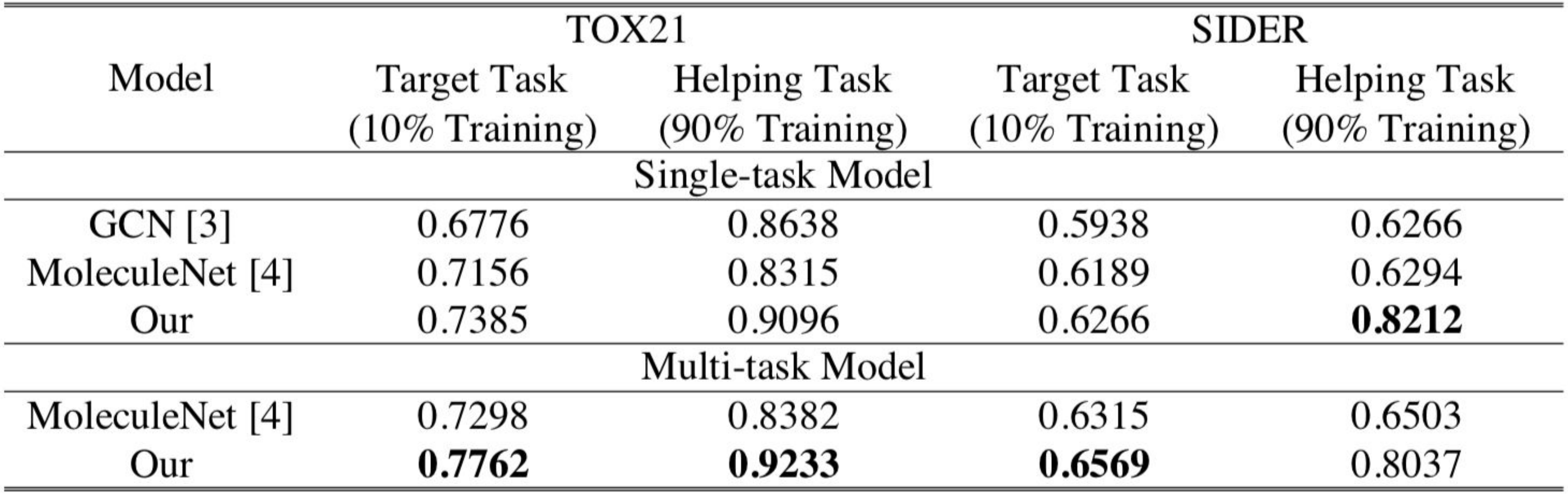
Target tasks (10% training):

- SR-ARE (TOX21)
- Investigations (SIDER)

Helping tasks: (90% training)

- SR-MMP (TOX21)
- Vascular Disorders (SIDER)

Experiment Results



Model	TOX21		SIDER	
	Target Task (10% Training)	Helping Task (90% Training)	Target Task (10% Training)	Helping Task (90% Training)
Single-task Model				
GCN [3]	0.6776	0.8638	0.5938	0.6266
MoleculeNet [4]	0.7156	0.8315	0.6189	0.6294
Our	0.7385	0.9096	0.6266	**0.8212**
Multi-task Model				
MoleculeNet [4]	0.7298	0.8382	0.6315	0.6503
Our	**0.7762**	**0.9233**	**0.6569**	0.8037

- Score will decrease if we lower the ratio of training data
- Score will increase if we leverage knowledge between tasks

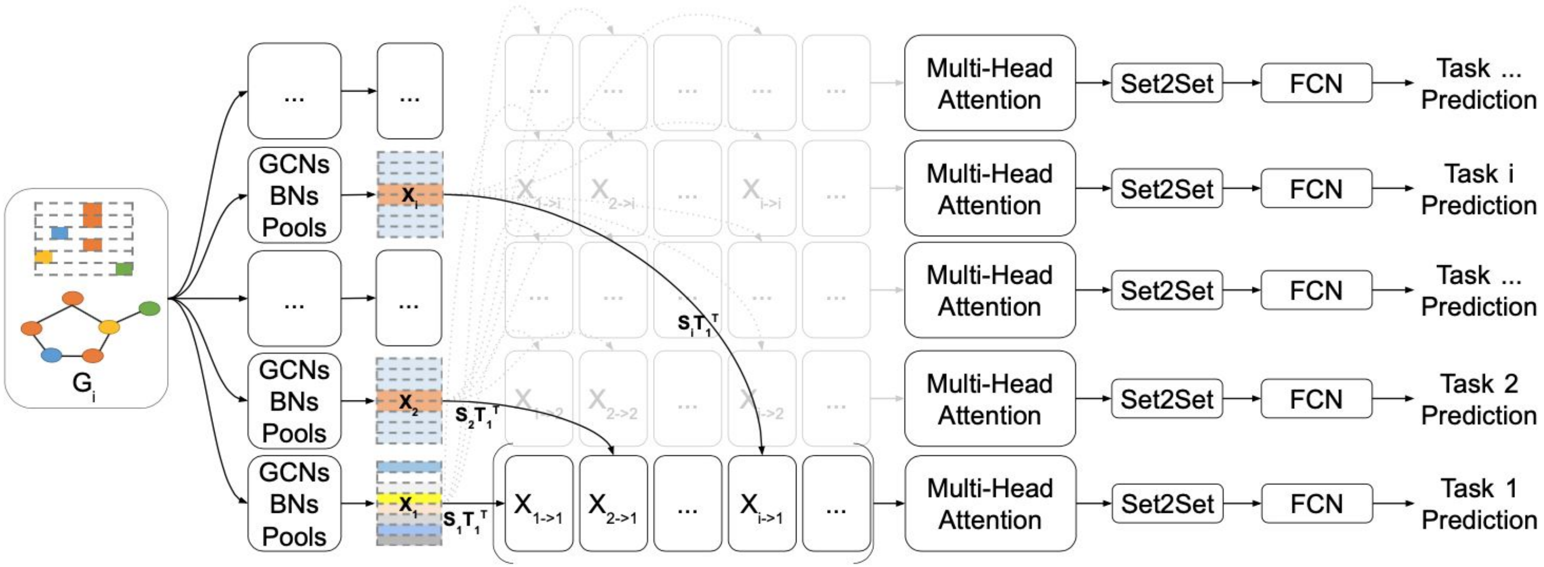
Z. Wu, B. Ramsundar, E. N. Feinberg; J. Gomes,C. Geniesse, A. S. Pappu, K. Leswing, V. Pande. MoleculeNet: A Benchmark for Molecular Machine Learning, 2017. https://arxiv.org/abs/1703.00564

Next Step

Backbone Model (Multi-Task)

- Problems of dual-task model
  - For every task pair (i, j), We need a transfer module $f_{i \to j}$
    - O(k^2), k is #tasks -> not scalable
  - Ignore the tasks relation at a higher level
- Insight:
  - Decompose the transfer module $f_{i \to j} = S_i T_j^{'}$
    - Each task i only need to store S_i and T_j
    - O(k), k is #tasks
  - Explicitly model task-level and graph-level relations

Backbone Model (Multi-Task)

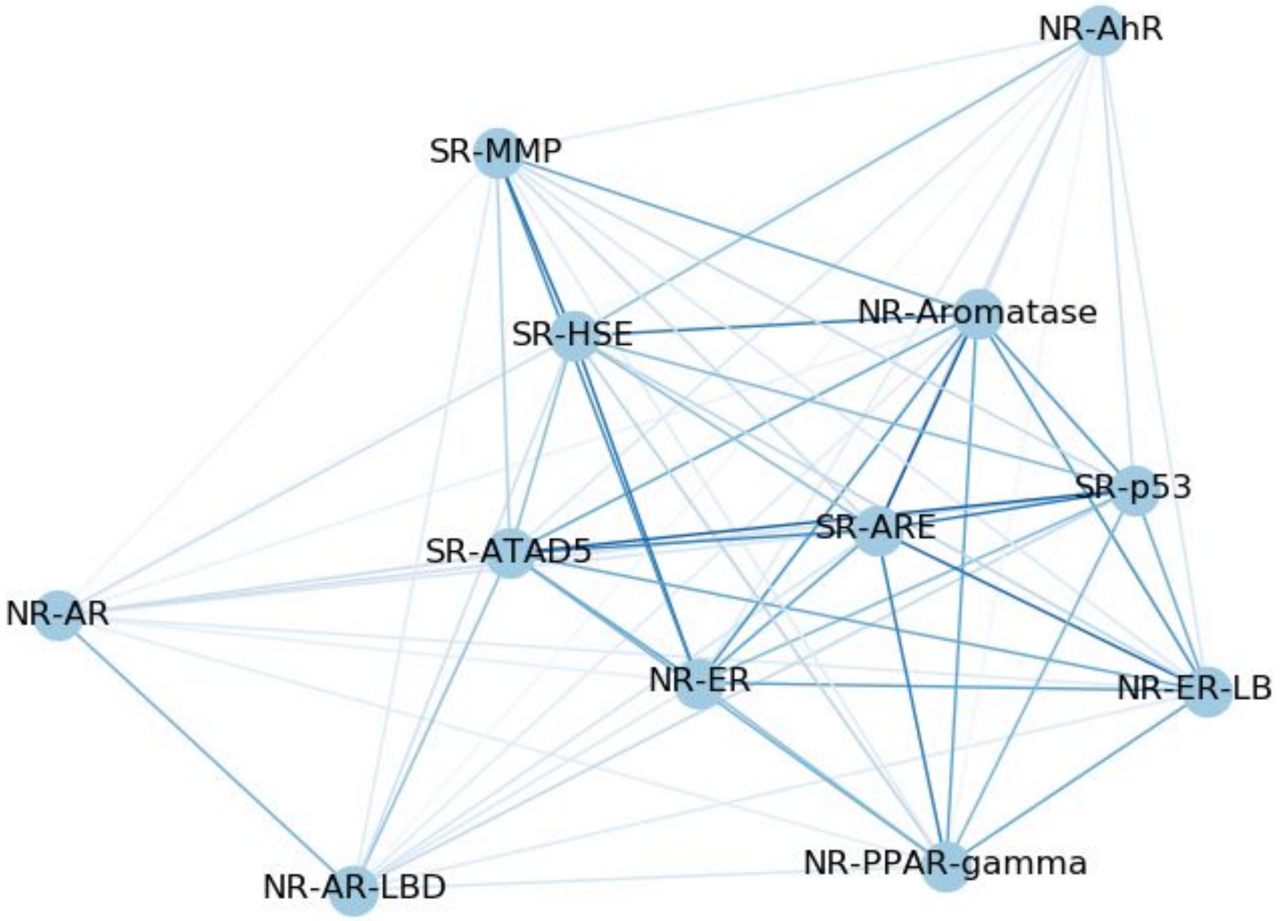


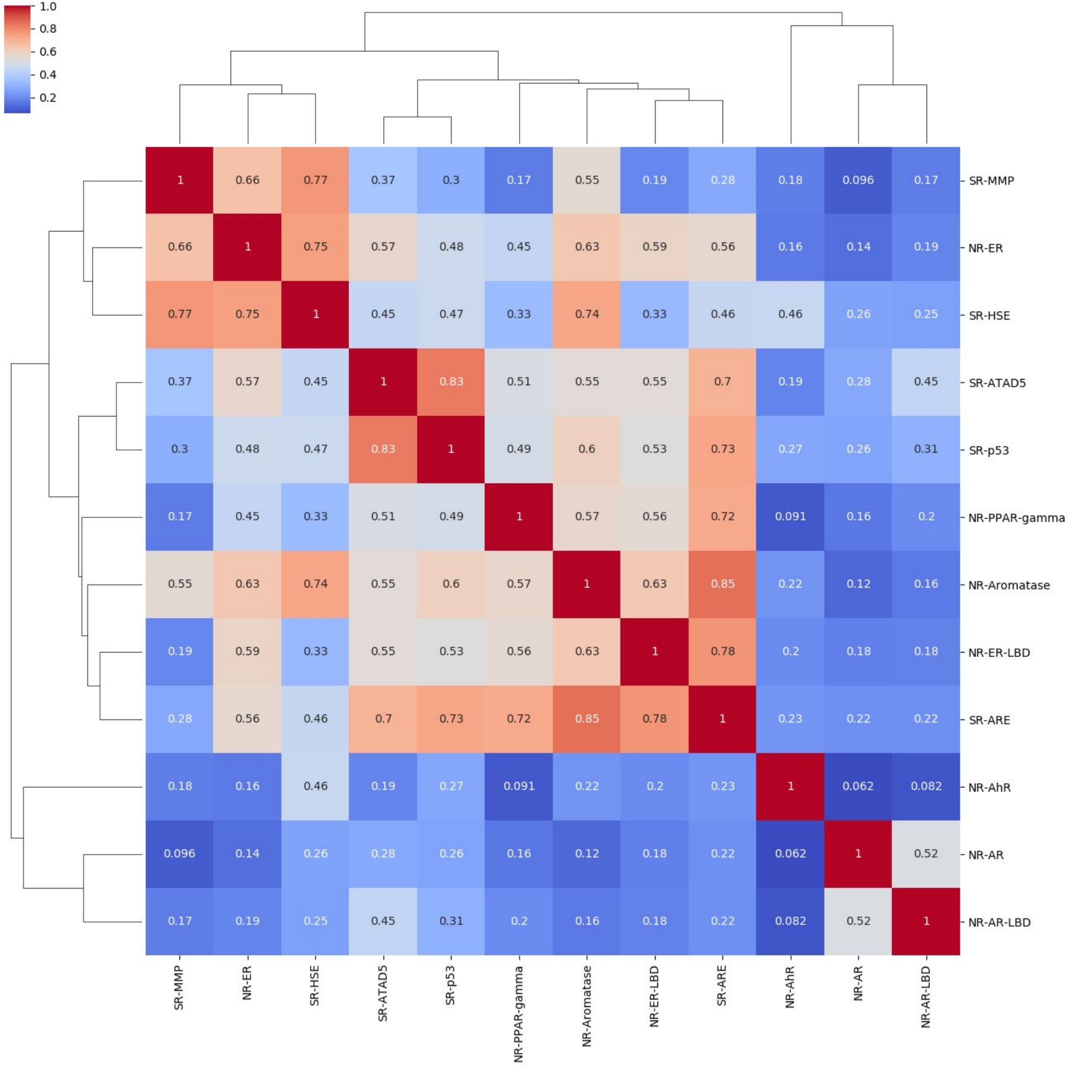
$$\hat{\boldsymbol{X}}_j = \Sigma_{i=1}^{k}[\mathrm{Softmax}(D_{i\to j}\boldsymbol{A}_{i\to j})\cdot \boldsymbol{X}_{i\to j}]$$

$\boldsymbol{X}_{i\to j} = \boldsymbol{X}_i\boldsymbol{S}_i\boldsymbol{T}_j^T$, where $\boldsymbol{S}_i, \boldsymbol{T}_j \in R^{d\times d'}$, $\boldsymbol{X}_{i\to j} \in R^{n\times d}$

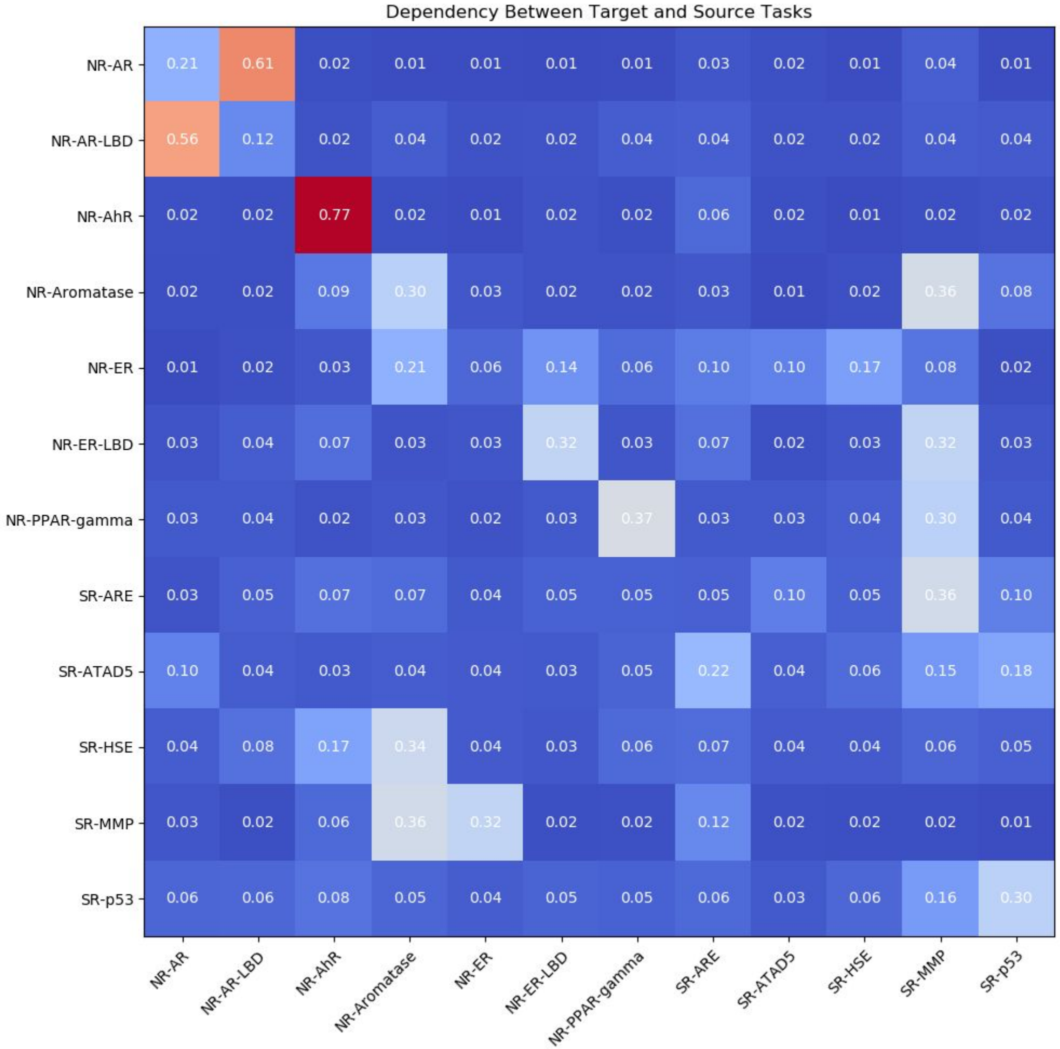
$\boldsymbol{A}_{i\to j} = \boldsymbol{X}_{i\to j}\boldsymbol{Q}_i(\boldsymbol{X}_j\boldsymbol{K}_j)^T$, where $\boldsymbol{Q}_i, \boldsymbol{K}_j \in R^{d\times d''}$, $\boldsymbol{A}_{i\to j} \in R^{n\times n}$

Some Results


NR-AhR
SR-MMP
NR-Aromatase
SR-HSE
SR-p53
SR-ARE
SR-ATAD5
NR-AR
NR-ER
NR-ER-LB
NR-PPAR-gamma
NR-AR-LBD



	SR-MMP	NR-ER	SR-HSE	SR-ATAD5	SR-p53	NR-PPAR-gamma	NR-Aromatase	NR-ER-LBD	SR-ARE	NR-AhR	NR-AR	NR-AR-LBD
SR-MMP	1	0.66	0.77	0.37	0.3	0.17	0.55	0.19	0.28	0.18	0.096	0.17
NR-ER	0.66	1	0.75	0.57	0.48	0.45	0.63	0.59	0.56	0.16	0.14	0.19
SR-HSE	0.77	0.75	1	0.45	0.47	0.33	0.74	0.33	0.46	0.46	0.26	0.25
SR-ATAD5	0.37	0.57	0.45	1	0.83	0.51	0.55	0.55	0.7	0.19	0.28	0.45
SR-p53	0.3	0.48	0.47	0.83	1	0.49	0.6	0.53	0.73	0.27	0.26	0.31
NR-PPAR-gamma	0.17	0.45	0.33	0.51	0.49	1	0.57	0.56	0.72	0.091	0.16	0.2
NR-Aromatase	0.55	0.63	0.74	0.55	0.6	0.57	1	0.63	0.85	0.22	0.12	0.16
NR-ER-LBD	0.19	0.59	0.33	0.55	0.53	0.56	0.63	1	0.78	0.2	0.18	0.18
SR-ARE	0.28	0.56	0.46	0.7	0.73	0.72	0.85	0.78	1	0.23	0.22	0.22
NR-AhR	0.18	0.16	0.46	0.19	0.27	0.091	0.22	0.2	0.23	1	0.062	0.082
NR-AR	0.096	0.14	0.26	0.28	0.26	0.16	0.12	0.18	0.22	0.062	1	0.52
NR-AR-LBD	0.17	0.19	0.25	0.45	0.31	0.2	0.16	0.18	0.22	0.082	0.52	1



Dependency Between Target and Source Tasks

	NR-AR	NR-AR-LBD	NR-AhR	NR-Aromatase	NR-ER	NR-ER-LBD	NR-PPAR-gamma	SR-ARE	SR-ATAD5	SR-HSE	SR-MMP	SR-p53
NR-AR	0.21	0.61	0.02	0.01	0.01	0.01	0.01	0.03	0.02	0.01	0.04	0.01
NR-AR-LBD	0.56	0.12	0.02	0.04	0.02	0.02	0.04	0.04	0.02	0.02	0.04	0.04
NR-AhR	0.02	0.02	0.77	0.02	0.01	0.02	0.02	0.06	0.02	0.01	0.02	0.02
NR-Aromatase	0.02	0.02	0.09	0.30	0.03	0.02	0.02	0.03	0.01	0.02	0.36	0.08
NR-ER	0.01	0.02	0.03	0.21	0.06	0.14	0.06	0.10	0.10	0.17	0.08	0.02
NR-ER-LBD	0.03	0.04	0.07	0.03	0.03	0.32	0.03	0.07	0.02	0.03	0.32	0.03
NR-PPAR-gamma	0.03	0.04	0.02	0.03	0.02	0.03	0.37	0.03	0.03	0.04	0.30	0.04
SR-ARE	0.03	0.05	0.07	0.07	0.04	0.05	0.05	0.05	0.10	0.05	0.36	0.10
SR-ATAD5	0.10	0.04	0.03	0.04	0.04	0.03	0.05	0.22	0.04	0.06	0.15	0.18
SR-HSE	0.04	0.08	0.17	0.34	0.04	0.03	0.06	0.07	0.04	0.04	0.06	0.05
SR-MMP	0.03	0.02	0.06	0.36	0.32	0.02	0.02	0.12	0.02	0.02	0.02	0.01
SR-p53	0.06	0.06	0.08	0.05	0.04	0.05	0.05	0.06	0.03	0.06	0.16	0.30

Conclusion

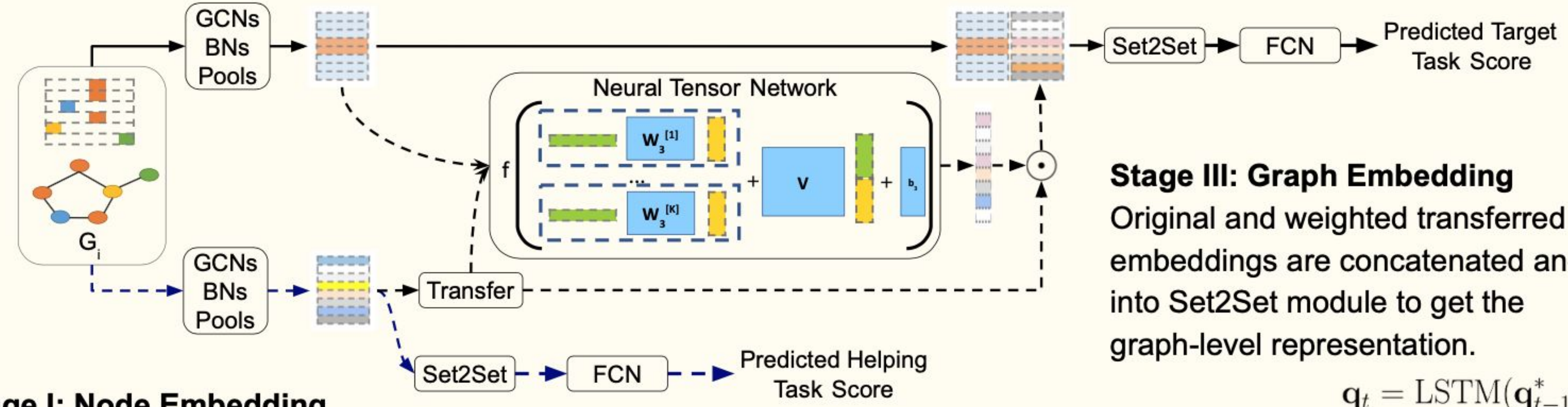
Conclusion

- Lack of labeled data and imbalanced label ratio limits the effect of DL in Chemistry and Biomedicine
- Contribution of our work: transfer + fusion
  - Introduce hypergraph to transfer knowledge between properties
  - Introduce novel attention mechanism to fuse transferred knowledge
  - Explore the hidden dependency structure between tasks
  - Improve dual-task’s AUC-ROC score by 6.9%

Q&A

- Introduction
- Related Work
  - Graph-Level Classification / Regression
  - Transfer Learning
  - Multi-Task Learning
- Method
  - Hypergraph Knowledge Transfer
- Experiment
- Next Step
- Conclusion

MODEL DETAIL



Stage III: Graph Embedding
Original and weighted transferred embeddings are concatenated and fed into Set2Set module to get the graph-level representation.

$$\mathbf{q}_t = \mathrm{LSTM}(\mathbf{q}^*_{t-1})$$
$$\alpha_{i,t} = \mathrm{softmax}(\mathbf{x}_i \cdot \mathbf{q}_t)$$
$$\mathbf{r}_t = \sum_{i=1}^{N} \alpha_{i,t}\mathbf{x}_i$$
$$\mathbf{q}^*_t = \mathbf{q}_t \,\|\, \mathbf{r}_t,$$

Stage I: Node Embedding

$$\mathrm{Conv}(A, X) = \hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}X\Theta$$
$$\mathrm{BN}(X) = \frac{X - \mathrm{E}[X]}{\sqrt{\mathrm{Var}[X] + \epsilon}} * \gamma + \beta$$
$$\mathrm{Pool}(v) = \max\{\max_{(u,v)\in E}\{u, v\}\}$$

Stage II: Knowledge Transfer
Two linear layers with ReLU activation transfer the node embeddings from helping task to target task.
Neural Tensor Network models the node-level interaction and decides the transfer weights.

Assume T_j is the target task.

Transfer the embeddings from T_i to T_j ($i \neq j$):

$X_{i \to j} = X_i W_{Si} W_{Tj}^T$, where $W_{Si}, W_{Tj} \in R^{d \times d'}$, $X_{i \to j} \in R^{n \times d}$.

Calculate the node-level attention:

$\mathrm{ATT}_{i \to j} = X_{i \to j} W_{Qi} (X_j W_{Kj})^T$, where $W_{Qi}, W_{Kj} \in R^{d \times d''}$, $\mathrm{ATT}_{i \to j} \in R^{n \times n}$.

$\widetilde{\mathrm{ATT}}_{i \to j} = \mathrm{Softmax}(\mathrm{ATT}_{i \to j}, \mathrm{dim} = -1) \in R^{n \times n}$.

Combine embeddings from all tasks w.r.t. node-level attention

$\hat{X}_{i \to j} = \widetilde{\mathrm{ATT}}_{i \to j} \cdot X_{i \to j}$ where $\hat{X}_{i \to j} \in R^{n \times d}$.

$X_j^{\mathrm{comb}} = \mathrm{cat}([X_j, \hat{X}_{: \to j}]) \in R^{k \times n \times d}$.

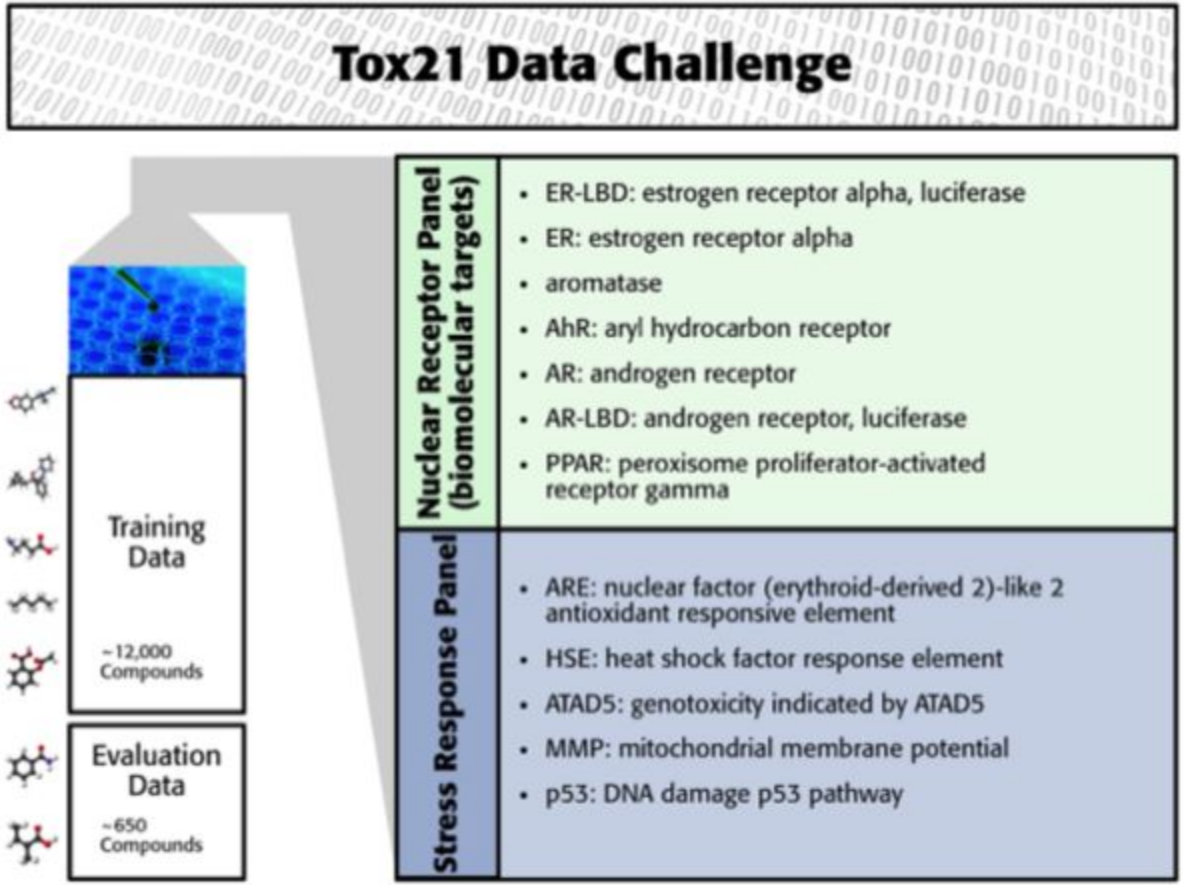
$\widetilde{X_j^{\mathrm{comb}}} = \mathrm{Norm}(X_j^{\mathrm{comb}}) \in R^{k \times n \times d}$.

Merge the embeddings w.r.t. task dependency:

$\hat{X}_j = W_{Dj} \widetilde{X_j^{\mathrm{comb}}}$, where $W_{Dj} \in R^{1 \times k}, \hat{X}_j \in R^{n \times d}$.

Finally, calculate graph-level embedding:

$G_j = \mathrm{Readout}(\hat{X}_j) \in R^{1 \times d}$.



Tox21 Data Challenge

Training Data

~12,000 Compounds

Evaluation Data

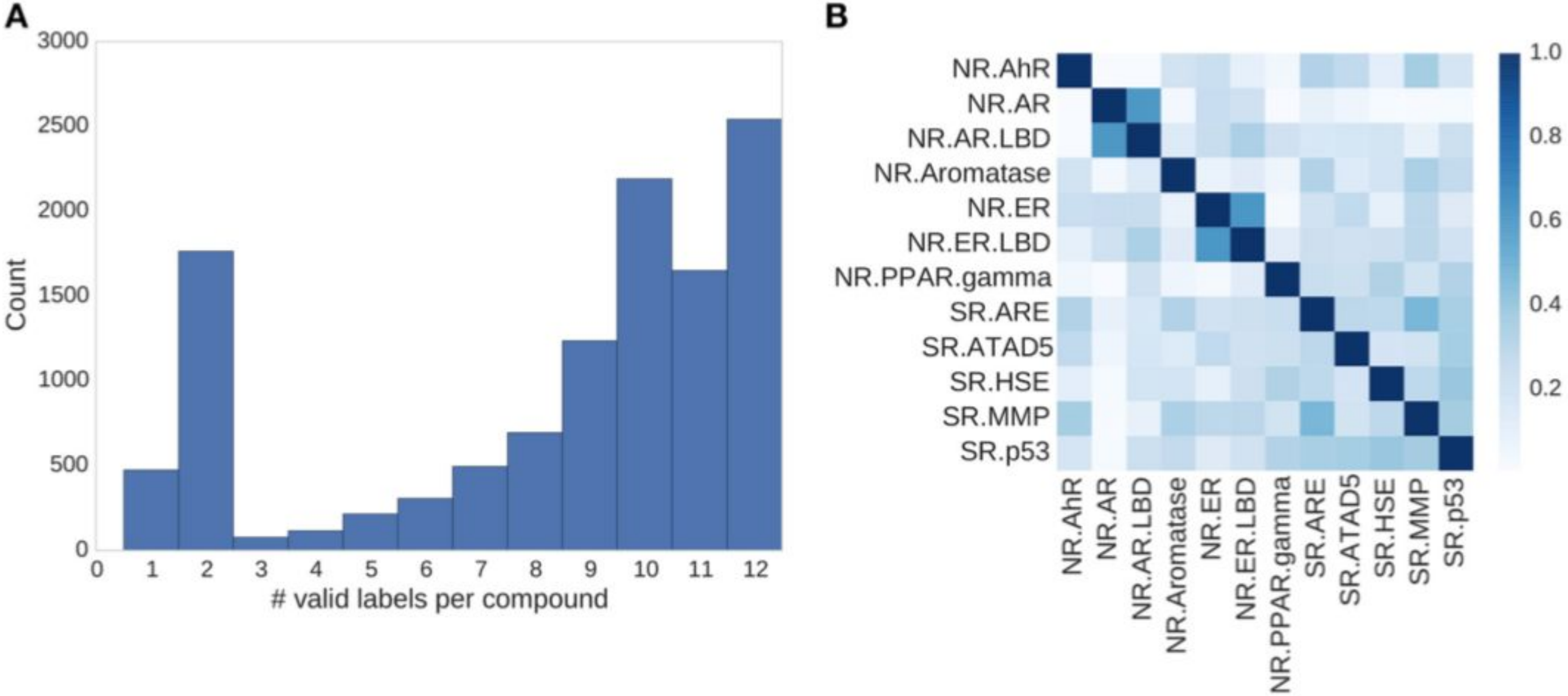
~650 Compounds

Nuclear Receptor Panel (biomolecular targets)

- ER-LBD: estrogen receptor alpha, luciferase
- ER: estrogen receptor alpha
- aromatase
- AhR: aryl hydrocarbon receptor
- AR: androgen receptor
- AR-LBD: androgen receptor, luciferase
- PPAR: peroxisome proliferator-activated receptor gamma

Stress Response Panel

- ARE: nuclear factor (erythroid-derived 2)-like 2 antioxidant responsive element
- HSE: heat shock factor response element
- ATAD5: genotoxicity indicated by ATAD5
- MMP: mitochondrial membrane potential
- p53: DNA damage p53 pathway


A
3000
2500
2000
1500
1000
500
0
Count
0
1
2
3
4
5
6
7
8
9
10
11
12
# valid labels per compound
B
NR.AhR
NR.AR
NR.AR.LBD
NR.Aromatase
NR.ER
NR.ER.LBD
NR.PPAR.gamma
SR.ARE
SR.ATAD5
SR.HSE
SR.MMP
SR.p53
1.0
0.8
0.6
0.4
0.2