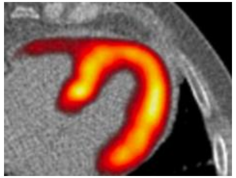
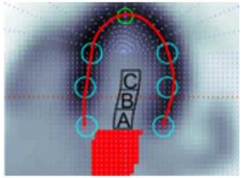
Who am I ?

LinkedIn

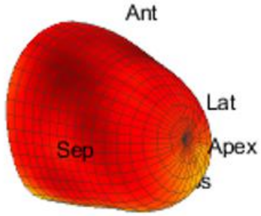
- My name is Arthur Chevalley
- 1st year PhD student in Life Science at UNIL and HES-SO Valais
  - EPFL bachelor in Micro-engineering and master in Robotics
  - Background in computer vision and various control strategies tasks
- Working on geometric models for cardiac tasks



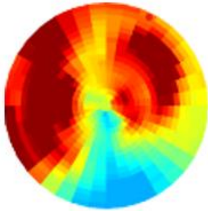
3D PET Reconstruction



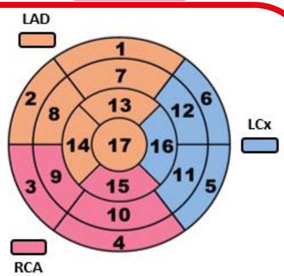
Centerline «segmentation»



3D Manifold/Mesh



Flattened polar map



Clinical Heart model

- Doctor working here for variety of tasks

→ Predict Major adverse cardiac events, heart tissues viability, microvascular diseases,.. using

- Multi-modal data and «population level» graphs
- 2D graphs and topological models (Cell complexes, Combinatorial Combination,..)
- 3D meshes/manifolds and topological models
- All of the above with time dependencies..

A quick intro of “Subgraph Neural Networks” & “GLASS: GNN with LAbeling tricks for Subgraph representation learning”

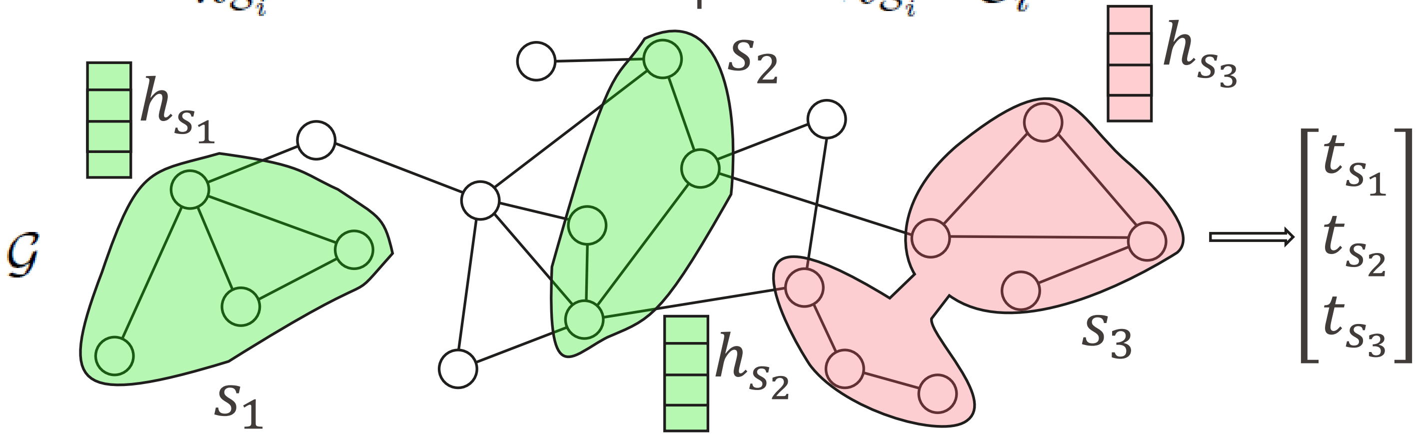
Graph representations for biology and medicine

Why Subgraphs ?

- Prediction concerning groups of node in a graph
- Need to consider topology inside and outside the subgraph
- Subgraphs have non-trivial internal structure, border connectivity, and notions of neighborhood and position relative to the rest of the graph
- Might consider multi-level information of nodes, edges, whole graph,..
- Either predict a subgraph target value or discover subgraphs

Some Grammar

- With a graph $\mathcal{G} = (\mathbb{V}, \mathbb{E}, \boldsymbol{X})$ with a finite node set $\mathbb{V} = \{1, 2, \dots, n\}$, and edge set $\mathbb{E} \subseteq \mathbb{V} \times \mathbb{V}$, and a feature matrix $\boldsymbol{X}$.
- A subgraph is defined as $\mathcal{S} = (\mathbb{V}_{\mathcal{S}}, \mathbb{E}_{\mathcal{S}}, X_{\mathcal{S}})$ with $\mathbb{V}_{\mathcal{S}} \subseteq \mathbb{V}$, $\mathbb{E}_{\mathcal{S}} \subseteq (\mathbb{V}_{\mathcal{S}} \times \mathbb{V}_{\mathcal{S}}) \cap \mathbb{E}$ and $X_{\mathcal{S}}$ the rows of $\boldsymbol{X}$ corresponding to the subgraph nodes
  - 3 subgraphs of 2 classes, 2 subgraphs with 1 CC and 1 subgraph with 2 CC
- Problem formulation: Given a graph $\mathcal{G}$, its subgraphs $\mathbb{S} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_n\}$ and their target properties $\boldsymbol{T} = \{t_{\mathcal{S}_1}, t_{\mathcal{S}_2}, \dots, t_{\mathcal{S}_n}\}$, the goal is to learn a representation $\boldsymbol{h}_{\mathcal{S}_i}$ that can be used to predict $\boldsymbol{t}_{\mathcal{S}_i}$ of $\mathcal{S}_i$.



subGNN [1]

- Reference for subgraph GNN
- Relies of a complex message-passing network:
  - Two components: Internal and Border
  - Three channels: Position (P), Neighborhood & (N) Structure (S):

	Interal components	Border components
P:	Distance between connected components of S	Distance between S and the rest of G
N:	Identity of internal nodes	Identity of border nodes
S:	Internal connectivity	Border Connectivity

- Requires the selection of anchor sets/points for each subgraph

channel	property	patch sampler	patch representation	similarity	
P	I	node in $\mathcal{S}$	node embedding	$1/(d(S^{(C)}, A)+1)$	Average shortest path
	B	node out of $\mathcal{S}$			
N	I	node in $\mathcal{S}^{(c)}$	node embedding	1	Average shortest path
	B	node in $N_k(\mathcal{S}^{(c)})$		$1/(d(S^{(C)}, A)+1)$	
S	I	connected components	rw_I + LSTM	$1/(dtw(S^{(C)}, A)+1)$	Dynamic time wrapping measure
	B		rw_B + LSTM		

subGNN - Detailed architecture

- Random sampling of anchor patches from G
  - Channel wise, i.e. P, N & S
  - Multiple anchor sets
- Message computed channel wise per subgraph components

$$\mathrm{MSG}_{\mathrm{X}}^{A\to S} = \gamma_{\mathrm{X}}(S^{(c)}, A_{\mathrm{X}}) \cdot \mathbf{a}_{\mathrm{X}}$$

Where γ_{X} is a similarity function, A_{X} the anchor set and $\mathbf{a}_{\mathrm{X}}$ the learned representation

- Message-passing

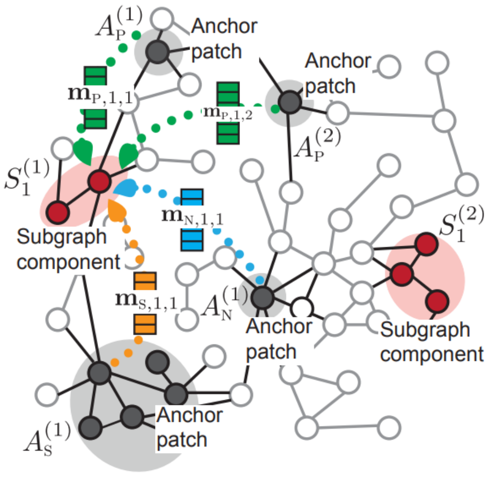
$$\mathbf{g}_{\mathrm{X},c} = \mathrm{AGG}_M(\{\mathrm{MSG}_{\mathrm{X}}^{A_{\mathrm{X}}\to S^{(c)}}\ \forall A_{\mathrm{X}} \in \mathcal{A}_{\mathrm{X}}\}),$$
$$\mathbf{h}_{\mathrm{X},c} \leftarrow \sigma(\mathbf{W}_{\mathrm{X}} \cdot [\mathbf{g}_{\mathrm{X},c}; \mathbf{h}_{\mathrm{X},c}]),$$

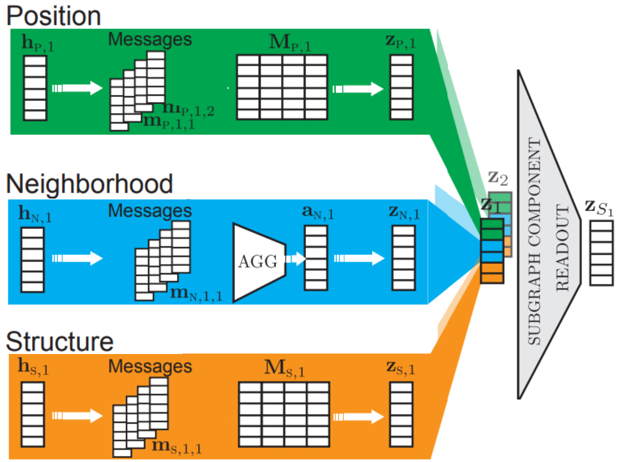
- Aggregate channel wise for a layer *l* and component *c*

$$\mathbf{z}_c^{(l)} = \mathrm{AGG}_C(\mathbf{h}_{\mathrm{N},c}^{(l)}, \mathbf{z}_{\mathrm{S},c}^{(l)}, \mathbf{z}_{\mathrm{P},c}^{(l)})$$

- Subgraph embedding

$$\mathbf{h}_S = \mathrm{READOUT}(\{\mathbf{z}_1, \ldots, \mathbf{z}_{n_c}\})$$

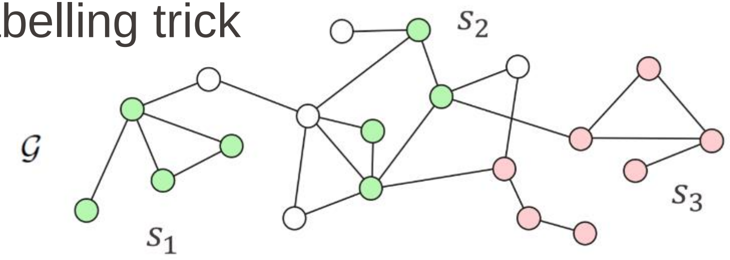




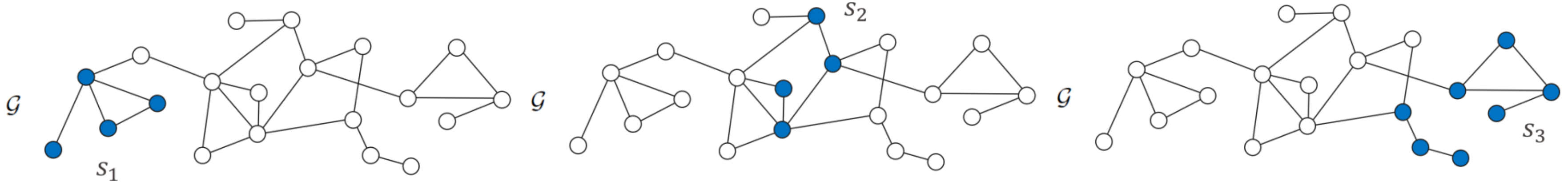
From original publication: https://arxiv.org/pdf/2006.10538

GLASS [2]

- Achieves subgraph tasks without complex MPNN scheme
- Learns the same properties as subGNN using node message-passing
- Relies on the labelling trick



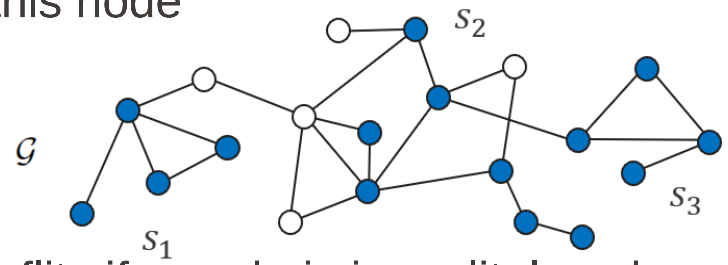
- Easiest labelling is the zero−one[3] defined as follow: $l_v^{(S)} = \begin{cases} 1 & if\, v \in \mathbb{V}_S \\ 0 & if\, v \notin \mathbb{V}_S \end{cases}$



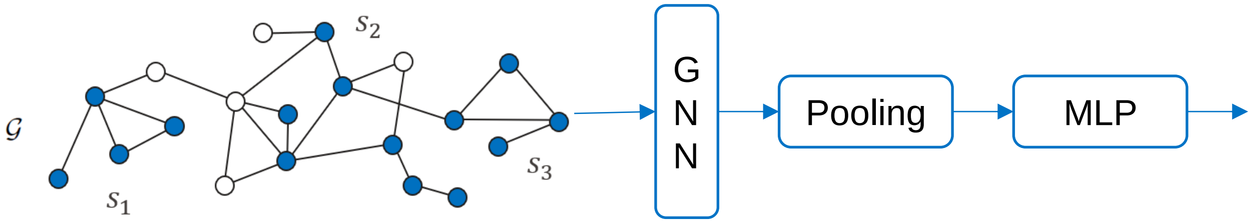
- Hard batch-training as input features vector changes with different subgraphs
- Longer training as message-passing is done $|S|$ times, i.e. each subgraph handled separetly

GLASS [2]

- Proposes a new labeling trick: max-zero-one
- Same idea as zero-one but gives a label 1 if at least one subgraph in the batch contains this node



- Might create conflits if a node is in mulitple subgraphs but:
  - Not if subgraphs are sparse
  - Provides regularization
- Extract node features with a pretrained self-supervised GNN



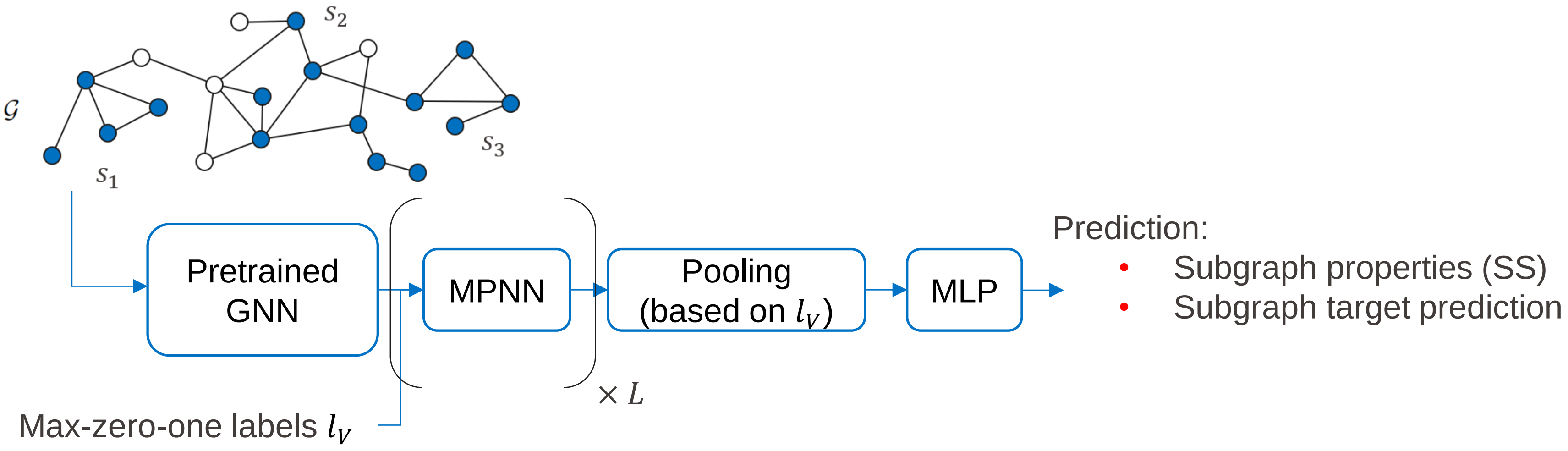
Prediction:

At node level:
- $A^k X$

At edge level:
- Graph link prediction

GLASS [2]

- Node embeddings from the SS pretrained GNN
- Message-passing on the whole graph, pool individual node embedding to produce multiple subgraphs.
- Computed embeddings are fed to a MLP for the task
- GLASS trained with a joint loss of subgraph structural properties prediction (cut ratio, coreness and #CC) and subgraph target prediction



Experiments

- Synthetic datasets:
  - Density: Tests internal structure
  - Cut ratio: Tests border structure
  - Coreness: Tests border structure and position
  - Components: Tests internal and external position

- Real datasets:
  - PPI-BP: Protein-Protein interaction network; biological processes are the subgraphs and predict their collective cellular function
  - EM-USER: Workout history of users; each user workout history is a subgraph and predict characteristics of that user
  - HPO-METAB/-NEURO: Phenotype and genotype information of rare disease; each subgraph is a rare disease and predict the metabolic or neurological disease

All datasets introduced in subGNN [1]

Results – Synthetic dataset

- Compare four models:
  - GLASS using node labeling trick and basic MPNN
  - *Classic* subGNN
  - Sub2Vec[4] which use RW to create subgraph features fed to an MLP (internal and external topology has no impact)
  - GNN-seg, a simpe MPNN for graph classification on each subgraph (external topology has no impact)

Method	density	cut ratio	coreness	component
GLASS	0.930 ± 0.009	$\mathbf{0.935 \pm 0.006}$	$\mathbf{0.840 \pm 0.009}$	$\mathbf{1.000 \pm 0.000}$
SubGNN	0.919 ± 0.006	0.629 ± 0.013	0.659 ± 0.031	0.958 ± 0.032
Sub2Vec	0.459 ± 0.012	0.354 ± 0.014	0.360 ± 0.019	0.657 ± 0.017
GNN-seg	$\mathbf{0.952 \pm 0.006}$	0.346 ± 0.011	0.593 ± 0.012	1.000 ± 0.000

- For cut ratio and coreness external topology neeeded
- Topology information needed for density and component

Results – Real dataset

- In addition to subGNN, Sub2Vec[4] and GNN-seg, two graph-agnostic methods using simple pooling to provide subgraph features

Method	ppi-bp	hpo-metab	hpo-neuro	em-user
GLASS	$\mathbf{0.619 \pm 0.007}$	$\mathbf{0.614 \pm 0.005}$	$\mathbf{0.685 \pm 0.005}$	$\mathbf{0.888 \pm 0.006}$
SubGNN	0.599 ± 0.008	0.537 ± 0.008	0.644 ± 0.006	0.816 ± 0.013
Sub2Vec	0.388 ± 0.001	0.472 ± 0.010	0.618 ± 0.003	0.779 ± 0.013
GNN-seg	0.361 ± 0.008	0.542 ± 0.009	0.647 ± 0.001	0.725 ± 0.003
MLP	0.445 ± 0.003	0.386 ± 0.011	0.404 ± 0.006	0.524 ± 0.019
GBDT	0.446 ± 0.000	0.404 ± 0.000	0.513 ± 0.000	0.694 ± 0.000

- Subgraph methods generally outperforms others
- Graph-agnostic methods are less good as graph structure is important
- Dense and localized graph require less external information
- Lower subgraph density gives clear benefits to subgraph methods

Personal thoughts

- Not suited for subgraph discovery as subgraphs are given as an input
- Not entierly convinced how GLASS performs with an important number of subgraphs in comparison to subGNN
- Interesting «easy to implement» idea to combine with other methods

Conclusion

- GLASS outperforms other approaches on subgraph tasks
- Conceptually simpler and easier to implement than subGNN
- GLASS is faster to train than subGNN
- Taking into account graph and subgraph structure is important

References

[1] Alsentzer, E., Finlayson, S., Li, M., & Zitnik, M. (2020). Subgraph neural networks. *Advances in Neural Information Processing Systems*, *33*, 8017-8029.

[2] Wang, X., & Zhang, M. (2021). GLASS: GNN with labeling tricks for subgraph representation learning. In *International Conference on Learning Representations*.

[3] Zhang, M., Li, P., Xia, Y., Wang, K., & Jin, L. (2021). Labeling trick: A theory of using graph neural networks for multi-node representation learning. *Advances in Neural Information Processing Systems*, *34*, 9061-9073.

[4] Adhikari, B., Zhang, Y., Ramakrishnan, N., & Prakash, B. A. (2018). Sub2vec: Feature learning for subgraphs. In *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part II 22* (pp. 170-182). Springer International Publishing.

Thank you for your attention

Questions time!

Feel free to contact me if you have any questions or suggestions about my project :

arthur.chevalley@unil.ch / LinkedIn

Applications of Subgraph GNN

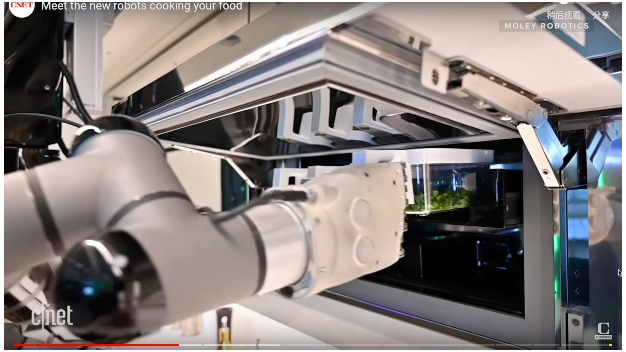
Yan Zhang

École
polytechnique
fédérale
de Lausanne

About me

- Robot Learning and Interaction Group @ Idiap; 3rd year PhD at EDEE
  - Research topic: Robot long-horizon manipulation planning

Table Rearrangement [1]



Kitchen Activities [2]

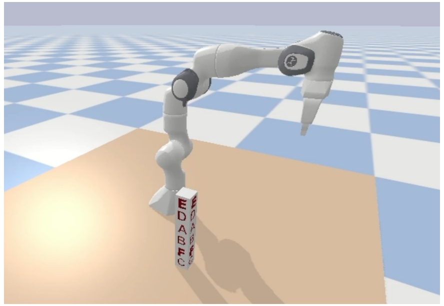
Long-horizon manipulation tasks in logistics and kitchen

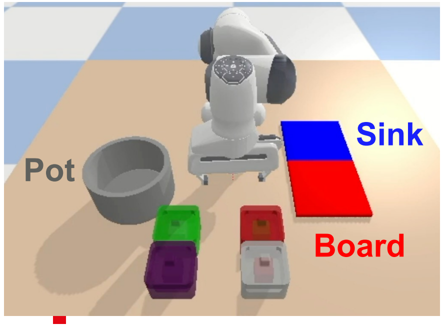
1. https://www.ocadogroup.com/
2. https://youtu.be/N3MpT3qeEGY

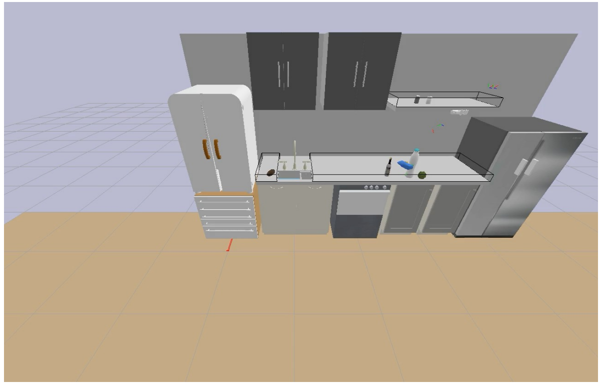
About me

- Robot Learning and Interaction Group @ Idiap; 3rd year PhD at EDEE
    - Research topic: How to build a robot chef

Given the name of my expect dish, it cooks autonomously







[Dong et al. *Soft fixtures*, 2023]

Subgraph-Aware Graph Kernel Neural Network for Link Prediction in Biological Networks

Research areas of this paper

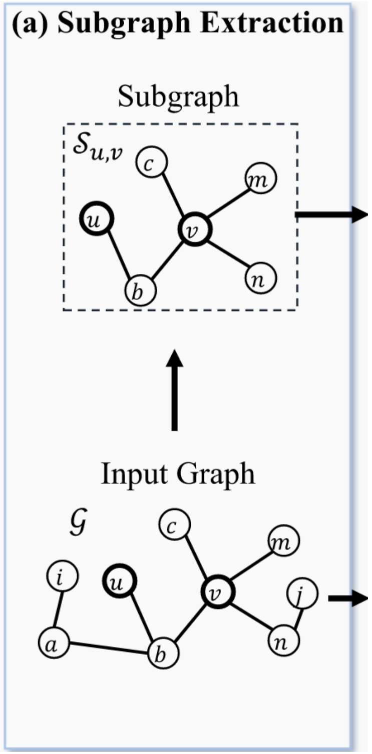
Biological network link prediction

1. **Microbe-disease association (MDA)**
   - ❑ Identify which microbes are associated with specific diseases;
2. **Drug-target interaction (DTI)**
   - ❑ Understand how drugs interact with their biological target for effective therapies;
3. **miRNA-disease association**
   - ❑ Understand how microRNAs are involved in the regulation of diseases;
4. **Drug-drug associations (DDAs)**
   - ❑ Discover novel functions of available drugs by the identification of DDAs;
5. **Protein-protein interactions (PPIs)**
   - ❑ Discover disease biomarkers through the identification of PPIs;

Methods

1. **Subgraph GNN:**
    1. each node may play a distinct role in different links of a network;
    2. SOTAs often ignore shared information among subgraphs, which are associated with biological functions.
2. **Graph Filters and Kernels:**
    1. calculate graph similarity by decomposing input graphs into atomic substructures;
    2. Therefore, can be used to capture shared information among subgraphs
3. **Diversity regularization:**
    1. Maximize the difference between graph filters
4. **Node embedding as auxiliary information**
    1. Help to differentiate node pairs that share the same subgraph

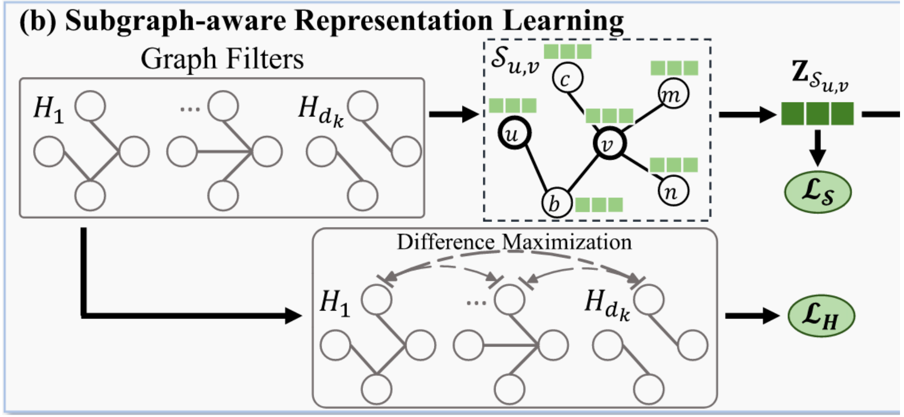
Methods- Subgraph Generation



Each subgraph $S_{u,v}$ corresponds to the topology of the h-hop neighborhood of two nodes u and v .

$$\mathcal{N}^h(u,v) = \{e | \min\left(dis(e,u), dis(e,v)\right) \leq h\}$$

Methods- Subgraph Aware Representation Learning



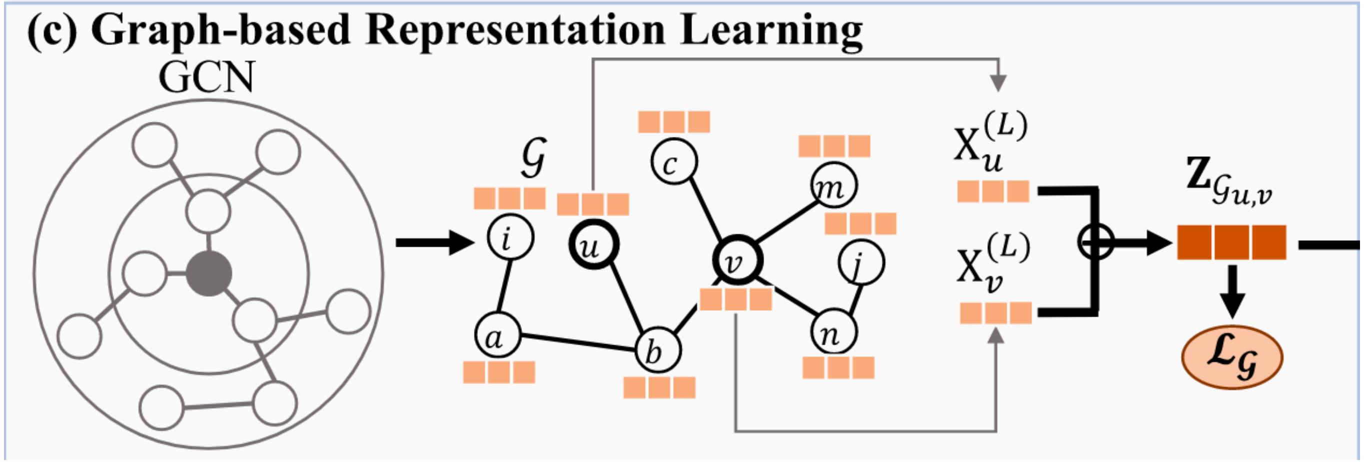
This block captures share information among subgraphs of node pairs:

1. Trainable Graph Filter with Difference Maximization;
2. Graph Kernel function (random walk) to pool node embeddings with different graph filters.

Graph filter

- Jointly trained while doing link prediction;
- With **Difference Maximization** to enforce the fixed set of **Graph Filters** to capture different perspectives of shared information;
- Done by penalizes graph filters that are close to each other with **L2 distance**

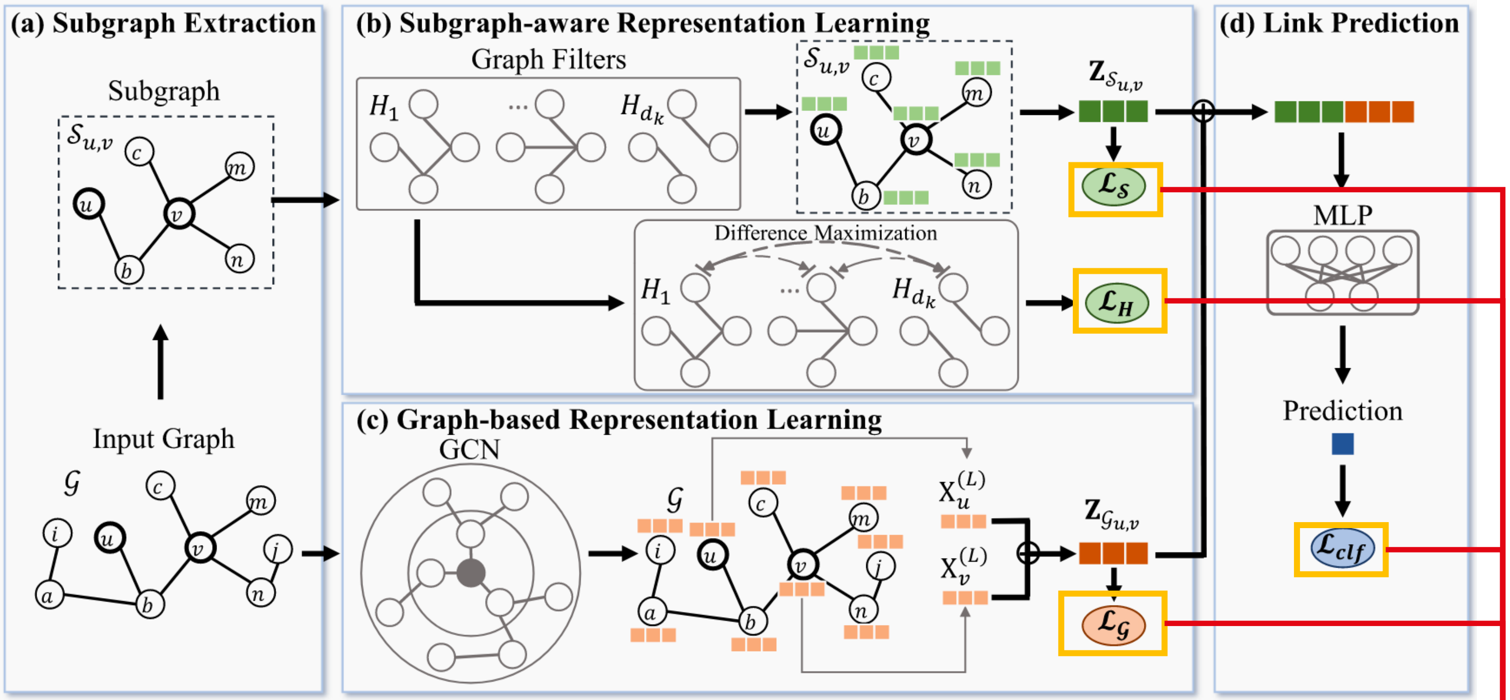
Methods- Auxiliary Information



GCN with original graph structure to update node embedding.

This block is designed to capture addition information that subgraph-aware GNN filtered out.

Pipeline



$$\mathcal{L} = \mathcal{L}_{clf} + \lambda_1 \mathcal{L}_H + \lambda_2 \mathcal{L}_{\mathcal{S}} + \lambda_3 \mathcal{L}_{\mathcal{G}}$$

Comparison with baselines

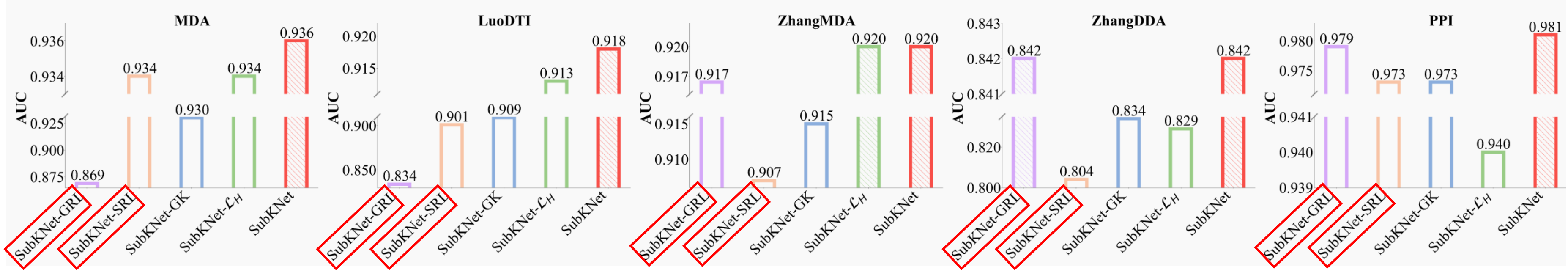
TABLE I
AUC Scores of SubKNet and Baselines on the Cross-Validation Sets

Categories	Methods	MDA	LuoDTI	ZhangMDA	ZhangDDA	PPI
Task-dependent methods	NinimHMDA	0.895±0.018	-	-	-	-
	MVGAEW	0.751±0.042	-	-	-	-
	IIFDTI	-	0.890±0.009	-	-	-
	GeNNius	-	0.915±0.010	-	-	-
	SFGAE	-	-	0.898±0.006	-	-
	CGHCN	-	-	0.912±0.005	-	-
	DRWBNCF	-	-	-	0.833±0.007	-
	RSML-GCN	-	-	-	0.817±0.005	-
	RAPPPID	-	-	-	-	0.887±0.010
	HNSPPI	-	-	-	-	0.914±0.007
Task-independent methods	SiGraC	0.866±0.024	0.704±0.013	0.787±0.010	0.669±0.008	0.564±0.009
	MVGCN	0.907±0.013	0.897±0.014	0.915±0.004	**0.845±0.004**	-
	LR-GNN	0.893±0.021	0.899±0.017	0.904±0.007	0.796±0.021	0.810±0.008
	CGCN	0.891±0.011	0.865±0.024	0.873±0.007	0.801±0.005	0.895±0.010
Subgraph-based methods	SEAL	0.888±0.017	0.903±0.013	**0.920±0.003**	0.827±0.004	0.869±0.008
	LGLP	0.891±0.016	0.904±0.012	0.918±0.004	0.807±0.006	0.879±0.009
	NNESF	0.856±0.017	0.839±0.017	0.753±0.013	0.694±0.004	0.877±0.009
	GCN-PS2	0.818±0.021	0.855±0.026	0.898±0.009	0.837±0.005	0.876±0.019
	SubKNet	**0.936±0.008**	**0.918±0.012**	**0.920±0.003**	0.842±0.007	**0.981±0.005**

Note: The bold in each column represents the highest score and the underlined denotes the second-best score. The standard deviation (±) is computed from 5-fold cross-validation results.

SubKNet exhibits superior performance on four out of five datasets, compare to task-dependent methods, task-independent methods, and subgraph-based methods

Ablation Study



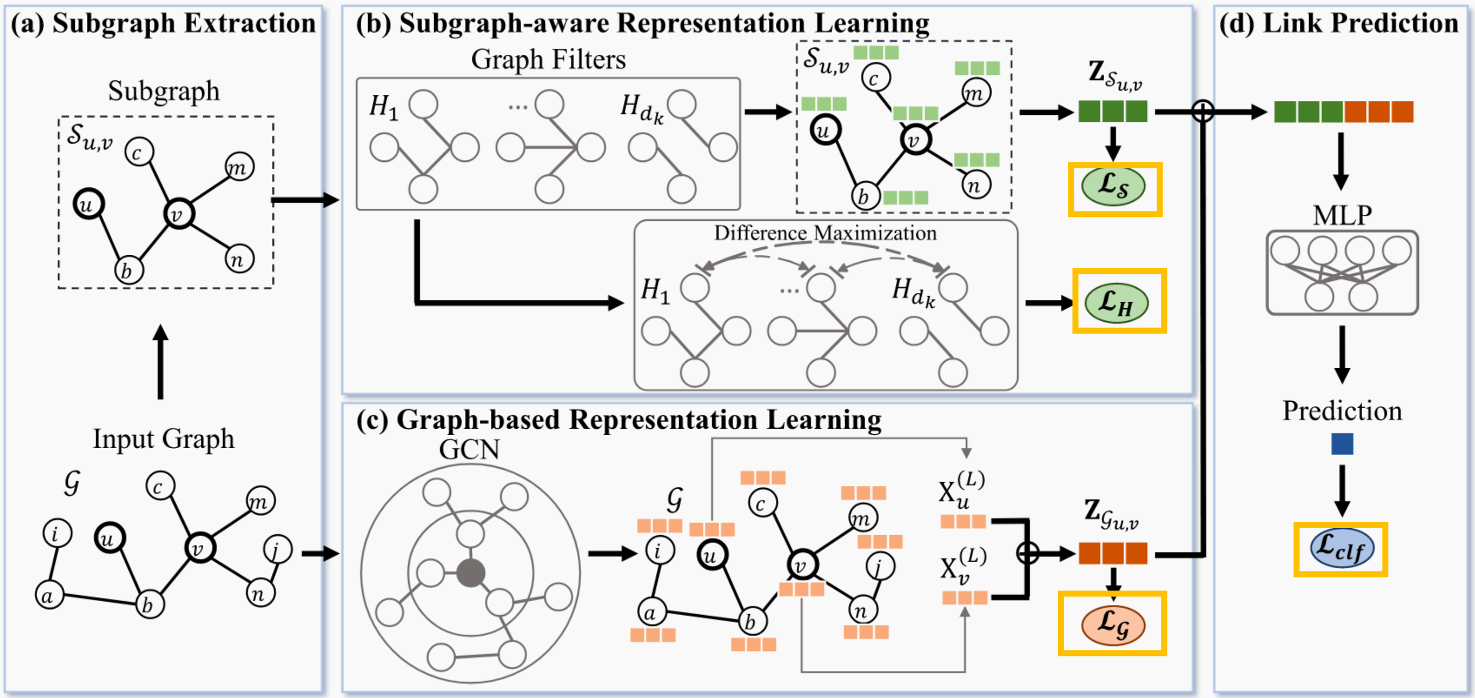
- SubKNet-GRL: no auxiliary information
- SubKNet-SRL: no subgraph-aware
- SubKNet-GK: vanilla SubGNN
- SubKNet-Lh: no diversity regularization

1. SubKNet VS (SubKNet-GRL & SubKNet-SRL): subgraph-aware and auxiliary information are important;
2. SubKNet VS SubKNet-GK: Graph Kernel and Filters;
3. SubKNet VS SubKNet-Lh: diversity regularization.

Conclusion & Limitation

- Subgraph-aware graph kernel NN (SubKNet), task-independent method, improves the link predication performance for biological networks;
- The main contributions are:
  1. Trainable subgraph-filters to capture shared information of nodes among subgraphs;
  2. Diversity regularization ensure graph filters capture different perspectives of shared information;
  3. Node embedding with original graph network as auxiliary information helps

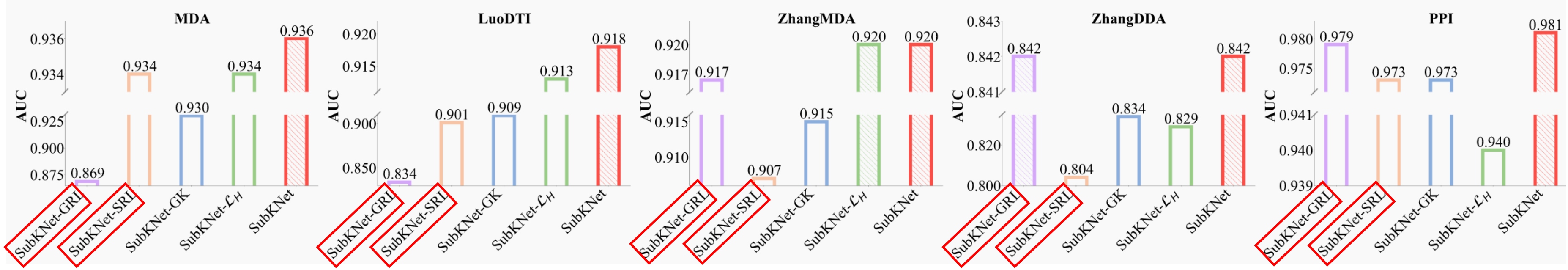
Personal Questions



Subgraph-aware -> shared information; Graph-aware -> other information

Wouldn't the Graph-based NN works well ?
How to define the subgraph structure ?

Personal Questions



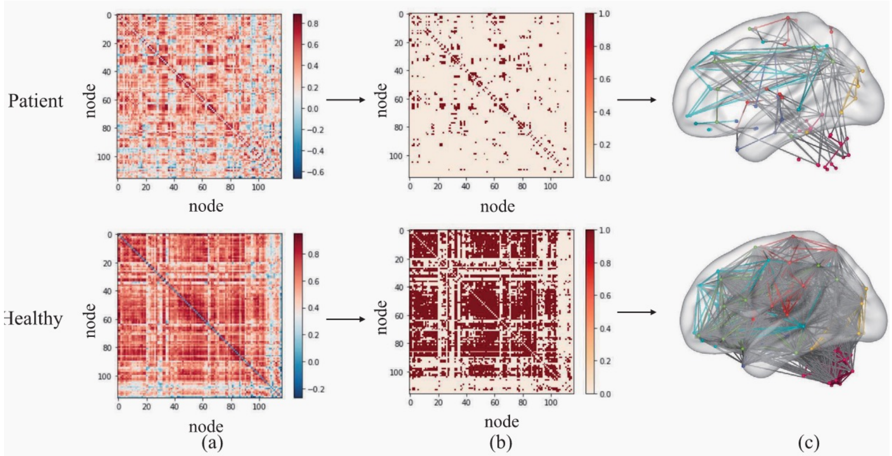
- SubKNet VS SubKNet-SRL (without subgraph-aware) actually shows not bad performance.
- The figures are actually tricky

Any questions?

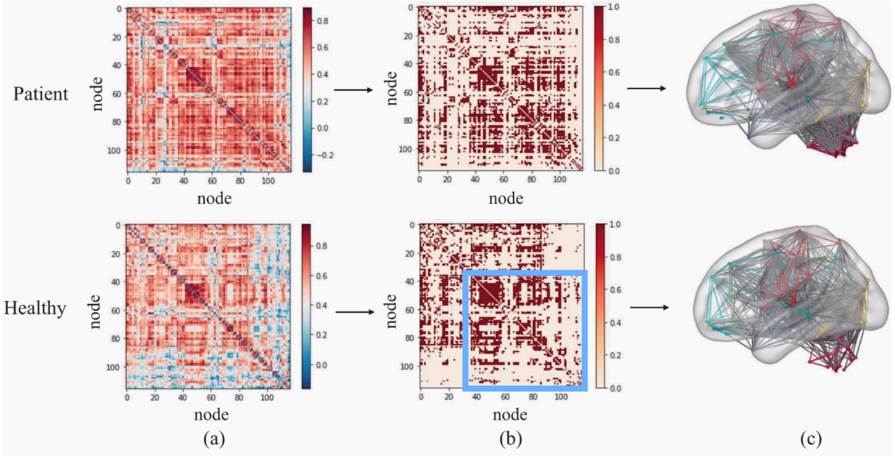
Knowledge Distillation Guided Interpretable Brain Subgraph Neural Networks for Brain Disorder Exploration

Research areas of this paper

Brain Disorder Exploration



Parkinson’s Disease (PD)

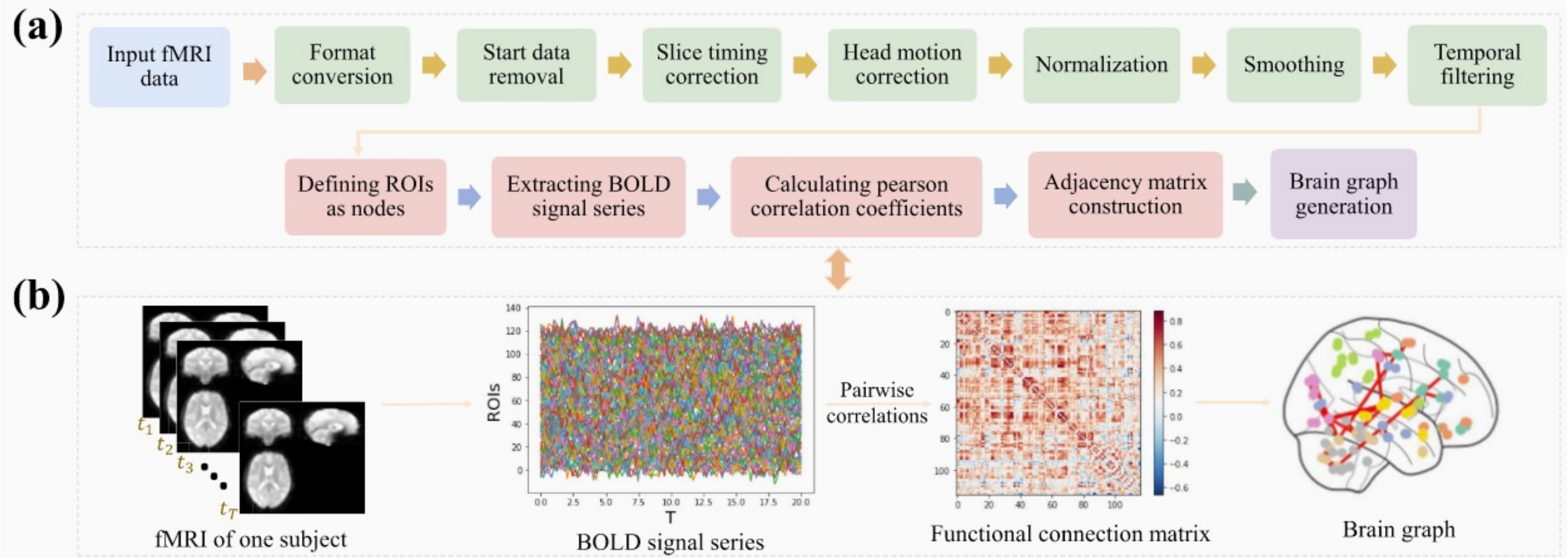


Attention-deficit/hyperactivity disorder(ADHD)

Methods

1. **Subgraph GNN:**
    1. The occurrence of a brain disorder involves not only damage to the corresponding functional brain, but also abnormal functional connections between these regions;
    2. SOTAs do not consider abnormal functional connection;
2. **Knowledge Distillation (KD) as guidance**
    1. Experiment data is small
    2. No enough clinical neuroimaging data + the storage, preprocessing, and brain graph modeling is expensive

Methods – Brain Graph Modeling based on fMRI data

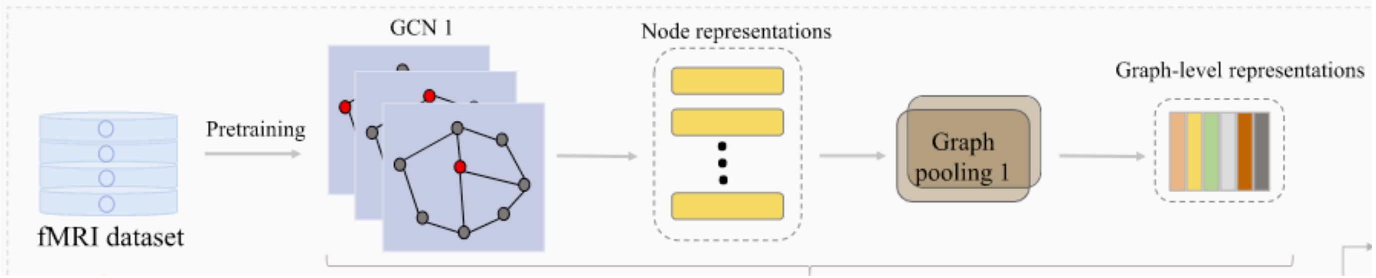


All brain graphs have the same number of nodes but they have different graph structures due to the unified brain region division G=(V, E, **W,** y)

W -> initial weighted functional connectivity matrix describing the connection strengths

$W_{i,j} > \lambda$ represents a connection between nodes

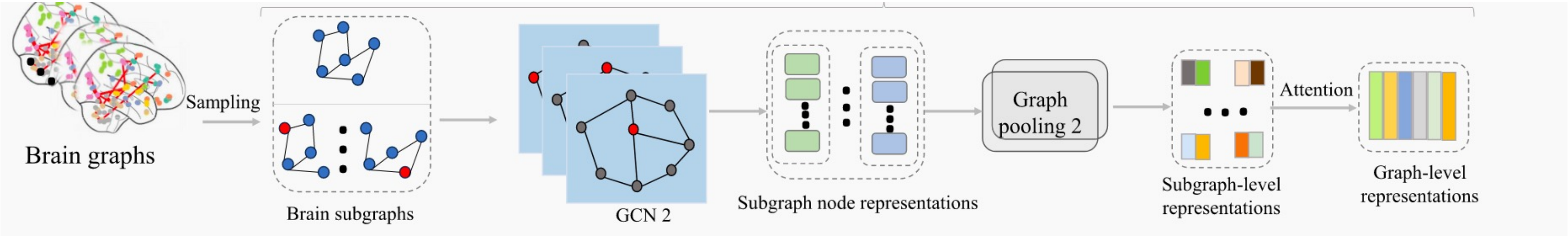
Methods – Pretrained teacher module



utilizing two-layer GCN model to pretrain teacher model on experimental brain graph data.

The output brain graph representations can be used as a prior knowledge to distill important subgraphs and are as global graph information to enhance graph-level representations

Methods - Subgraph distillation module



- One-layer GCN for subgraph node representation learning

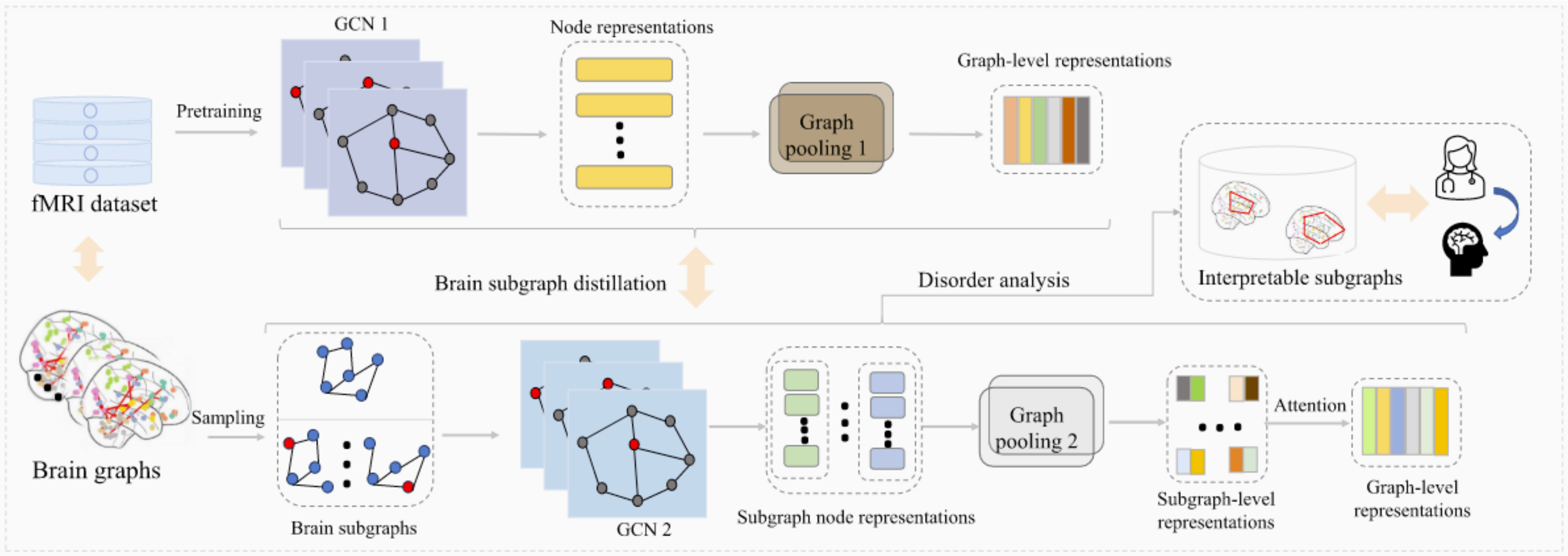
$$q^i = \mathbf{Z}_{\text{sub}}^i \cdot \mathbf{W}_{\text{sub}} \cdot \mathbf{H}_{\text{teacher}} \quad s^i = \frac{\exp(q^i)}{\sum_{j=1}^n \exp(q^j)} \quad \mathbf{H}_{\text{sub}} = \sum_{j=1}^n s^j \mathbf{Z}_{\text{sub}}^j \quad \mathcal{L}_{\text{mse}} = f_{\text{mse}}(\acute{y}, \bar{y})$$

Top-*k* discriminative subgraphs are selected

$$\mathcal{L}_{\text{KD}} = \text{KL}(p^s, p^t) \quad \mathcal{L} = \mathcal{L}_{\text{cro}} + \beta_1 \mathcal{L}_{\text{mse}} + \beta_2 \mathcal{L}_{\text{KD}}$$

Such loss is designed to capture both local and global graph information to enhance prediction performance

Pipeline



Two-layer GCN learn a teach graph-level representation beforehand
For loop: 1) One-layer GCN learn a subgraph-level representation;
2) top-k subgraphs fused;
3) enrich pretrained teacher model (minimizing the KL loss);
4) back propagation.

EPFL

idiap RESEARCH INSTITUTE

Experiments

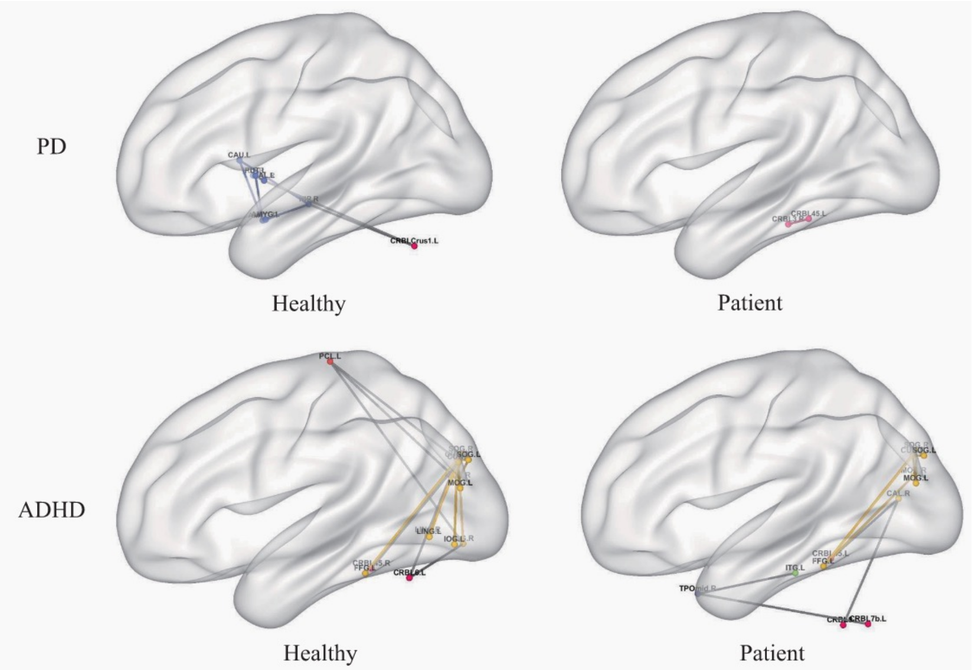
TABLE I

BRIAN GRAPH CLASSIFICATION RESULTS ABOUT THE MEAN ± STANDARD DEVIATION (%) ON TWO DATASETS

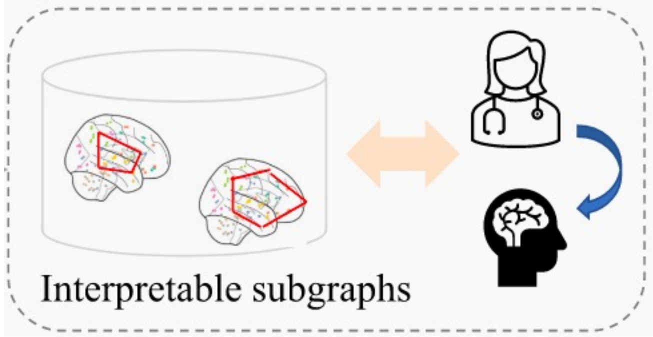
Methods	PD			ADHD		
	Accuracy	AUC	F1-score	Accuracy	AUC	F1-score
GCN	62.78±4.06	65.22±14.51	38.53±1.57	71.66±7.94	76.03±8.77	71.37±7.93
GAT	62.78±4.06	58.89±14.72	38.53±1.57	70.94±3.50	75.41±1.52	70.63±3.55
GIN	62.78±4.06	62.78±13.43	40.17±3.61	70.32±6.97	72.86±5.91	70.00±6.88
SortPool	65.28±6.33	65.67±8.14	45.50±14.00	70.97±7.71	76.91±7.18	70.30±8.54
SIB	65.28±6.33	59.56±17.74	47.03±10.65	69.52±6.77	70.88±8.41	69.08±6.95
BrainGNN	65.00±7.32	64.33±12.63	49.17±13.93	64.92±3.52	78.36±4.92	63.62±4.26
IBGNN	63.06±7.33	61.00±19.68	44.48±11.26	72.25±7.07	73.31±5.33	72.20±7.12
GCN-ours	65.28±6.33	**66.44±15.84**	44.17±11.36	**75.75±5.50**	80.35±6.73	**75.36±5.60**
GAT-ours	**67.22±8.98**	65.89±16.02	**49.67±14.88**	72.21±6.99	**80.78±5.64**	72.10±7.15
GIN-ours	62.78±4.06	62.89±8.21	41.38±4.36	72.25±9.33	69.73±11.29	72.16±9.29

Proposed methods work best compare with baselines

Experiments



With trained model, they can select the discriminative subgraphs for each patient subject based on attention scores, and visualize the subgraphs for disorder analysis



It shows similar correct results with classical methods: fALFF + DPARSF tool

EPFL

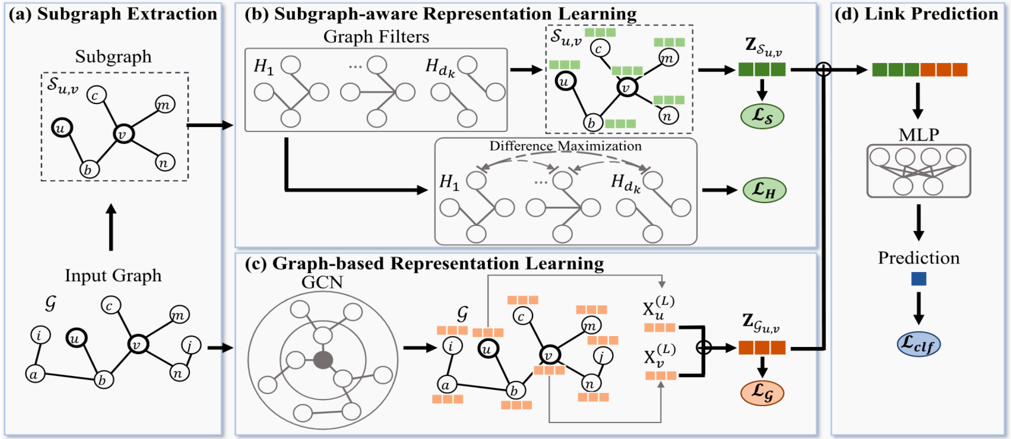
idiap RESEARCH INSTITUTE

Conclusion & Limitation

Contributions:

1. a complete process from brain graph modeling to brain disorder prediction and pathogenic analysis;
2. the first to introduce brain subgrpah view for disorder analysis.
3. propose KD guided brain subgraph neural networks to extract discriminative brain subgraphs under limited brain graph data.

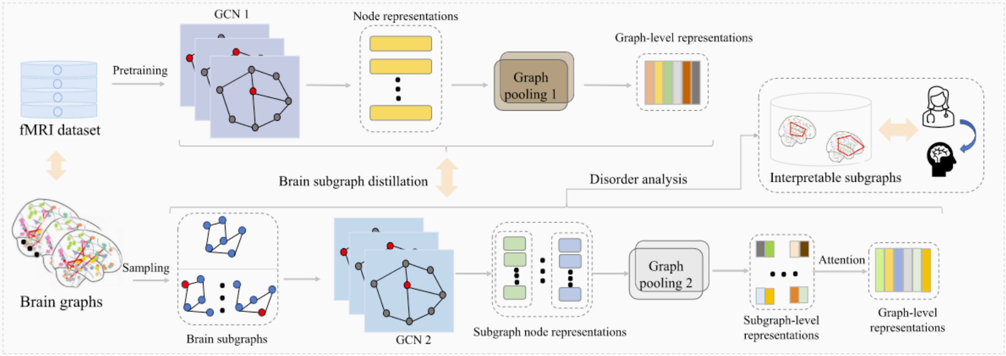
Personal Discussions



Use subgraph to improve link prediction performance

Neighborhood of node-pair as subgraph

Subgraph structure improves prediction performance



Find important subgraphs for diagnosis

Neighborhood of each node as subgraph

Thanks for your attention