3.5 Graph Neural Networks in Biomedicine

Petro Lio, Paul Scherer, Emma Rocheteau, Andreea Deac, Nikola Simidjievski
pl219@cam.ac.uk
https://www.cl.cam.ac.uk/teaching/2122/L45/

General tour of GNN

- Mathematical Formulation
    - Graphs: nodes (objects) + edges (interactions) between them
- Types of graphs
    - Undirected
    - Weighted
    - Multirelational
    - Heterogeneous
- Euclidean to non-Euclidean perspective
    - Increased interpretability
    - Better concentration of information
    - E.g. Superpixel
- Tasks on a graph: by scale
    - Node-level labels
    - Edge-level labels
    - Graph-level labels
    - Using neural network - use appropriate loss function
        - E.g. cross-entropy for classification or mean-squared error for regression
        - Optimise by gradient + backpropagation
    - Existing libraries:
        - PyTorch Geometric,
        - DGL
        - Spektral,
        - Jraph
- Formalization of Graph embedding
- Encoder-decoder Setup
    - Encode node
    - Decode neighborhood
- How a GNN operates
    - Input: Feature matrix X, preprocessed adjacency matrix A
    - Output:
        - Node classification:
            - softmax(z)
            - E.g. Kipf & Welling (ICLR 2017)
        - Graph classification:
            - softmax(sum z)
            - E.g. Duvenaud et al. (NIPS 2015)
        - Link prediction:
            - p(A) = sig(zz)

- Cancer Graph Network
    - https://www.medrxiv.org/content/10.1101/2021.09.01.21262086v2
- Multilayer networks
    - Not so much different to other networks
    - Adjacency matrix as a single matrix
- Directional graph neural networks
    - Grid Graph
    - Molecular graph
- Higher Order networks
    - Hypergraph
    - Multi-to-multi nodes
    - https://www.nature.com/articles/s41567-021-01371-4
- Systemic level signal detection
    - Multi-organ, multi-omics
    - Modelling connection between layer is complex
    - Hypergraph
        - https://www.biorxiv.org/content/10.1101/2022.07.31.502211v3
- Sheaf Neural Networks
    - https://arxiv.org/pdf/2206.08702.pdf
    - https://arxiv.org/pdf/2202.04579.pdf

Towards understanding mechanisms: adding logics

- https://arxiv.org/pdf/2007.04612.pdf

ODEs and deep learning

Digital Patient

- https://arxiv.org/pdf/2009.08299.pdf
- Better understanding systemically

From MLPs to GNNs for learning in biomedical contexts

Geo2DR: https://github.com/paulmorio/geo2dr
PyG-T: https://github.com/benedekrozemberczki/pytorch_geometric_temporal

Predicting Patient Outcomes - EHR in ICU

- Electronic Health Records in the ICU
    - https://arxiv.org/abs/2101.03940
- Distribution of diagnoses in eICU database
    - Mean frequency with each diagnosis in data – 229 patients
        - Not sufficient for DL algorithm
    - Traditionally
        - Only use common diagnoses – excluding rare diseases

        - Encoding layer to reduce dimension
    - Adopted to doctor's
        - How to learn from rare patient cases from doctors
            - Relatedness
- Relatedness
    - Doctor recalling how they treated rare cases in the past
        - Compare how new patient is similar and different from the past
        - Emulated to architecture
    - New representation for each patient
- Graph construction
    - Word embeddings from word2vec/ LightBERT followed by cosine similarity
    - New similarity matrix
        - Constructed with other clinicians

Diagnosis Graph Construction We start by assigning a pairwise similarity score between all patients. First, we transform the diagnoses into a multi-hot vector for each patient, resulting in a diagnosis matrix $\mathcal{D} \in \mathbb{R}^{N \times m}$ where m is the number of unique diagnoses and N is the number of patients. The similarity score $\mathcal{M}_{ij}$ between nodes i and j is defined as

$$\mathcal{M}_{ij} = a \overbrace{\sum_{\mu=1}^{m} \left(\mathcal{D}_{i\mu} \mathcal{D}_{j\mu} (d_\mu^{-1} + c) \right)}^{\text{Shared Diagnoses}} - \overbrace{\sum_{\mu=1}^{m} \left(\mathcal{D}_{i\mu} + \mathcal{D}_{j\mu} \right)}^{\text{All Diagnoses}} \quad (1)$$

where d_μ is the occurrence of diagnosis μ, and a and c are tunable constants. The first term positively rewards shared diagnoses. Note that the d_μ^{-1} term incorporates the idea that two patients sharing a rare diagnosis is more significant than a common one. The second term penalises the total number of diagnoses – this is to prevent patients with many diagnoses becoming 'hubs' of high connectivity, attracting imprecise matches with several non-shared diagnoses. We examine $\mathcal{M}$ under a k-Nearest Neighbour (k-NN) scheme to establish k edges per node. The parameters a, c and k were treated as hyperparameters ($c = 0.001$, $a = 5$ and $k = 3$ in the final model).

We experimented with alternative graph construction methods, such as applying a score threshold for edges (more akin to Malone et al. [18]), or using BERT [5] to encode diagnosis texts prior to similarity computation. Empirically we have found the presented method to work best through manual inspection of the resultant graph and preliminary experimentation.

- 1st dimension: index of patient
- 2nd dimension: index of diagnosis
- D = long diagnosis factors
- Taking *k* nearest neighbour to construct graph

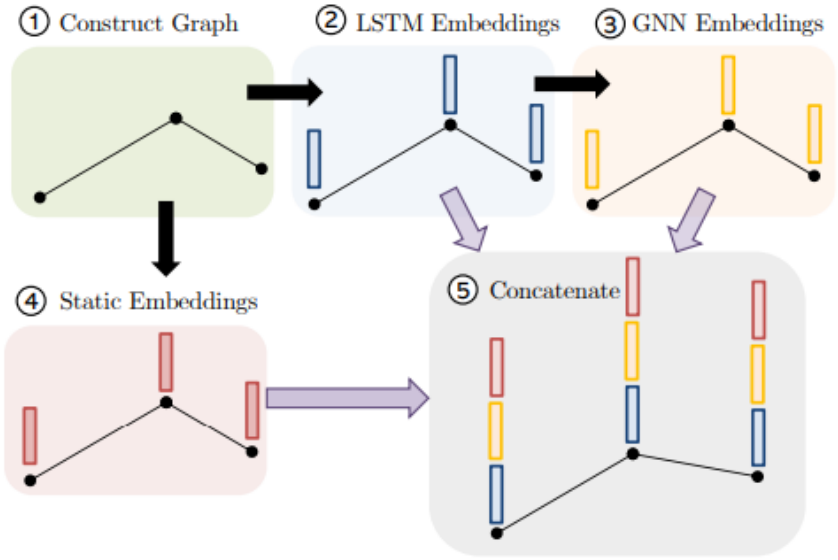

Fig. 2 Approach Overview. First, we construct a patient graph, which becomes the input to an end-to-end LSTM-GNN model. Through the LSTM-GNN, each node's temporal features are encoded by a temporal encoder followed by a graph encoder, and the static features are encoded separately. Finally, these are concatenated and passed to a fully-connected layer for prediction.

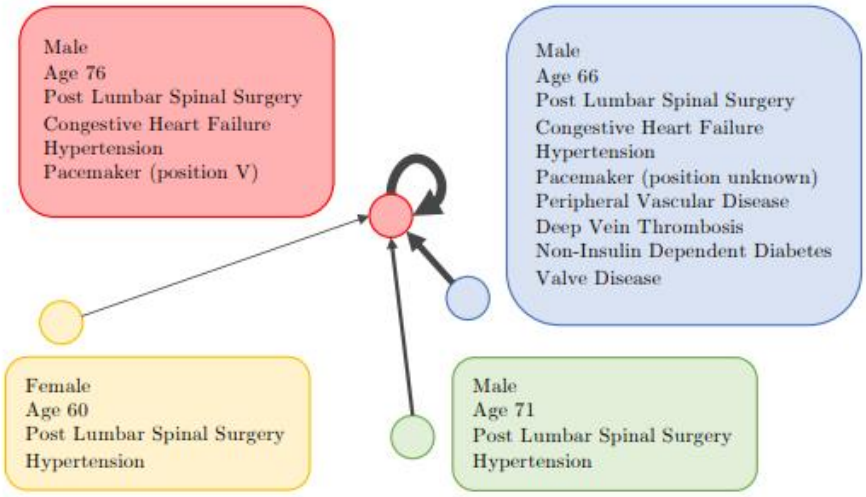

Fig. 4 An example showing graph attention weights in LSTM-GAT* (indicated by edge thickness).

Inductive biases for molecular interactions

- Molecular interactions: Infer the bipartite links between graphs
    - Antibody-antigen interactions
        - https://arxiv.org/abs/1806.04398
        - https://arxiv.org/abs/2106.00757
- Motivation for studying antibodies
    - Antibodies are

        - Y-shaped proteins
        - Critical part of our immune system
    - Neutralise pathogenic bacteria and viruses by tagging the antigen
        - “Lock and key” system
    - Designing antibodies would be a big step towards personalised medicine
- Towards personalised medicine
    - Generating an antibody requires first predicting the specific amino acids (the paratope) which participate in the neutralisation of the antigen
    - Input:
        - A sequence of (one-hot encoded) antibody amino acids
        - +) A sequence of (one-hot encoded) antigen amino acides
    - Output:
        - Probability for each amino acid to participate in binding with the antigen

Common challenges in practice: real data from clinical cancer trials

- Multimodal and high dimensional data
    - Complex models dont scale nor transfer
    - Sophisticated approaches need substantial amount of data (samples) for training
- OntoGCN

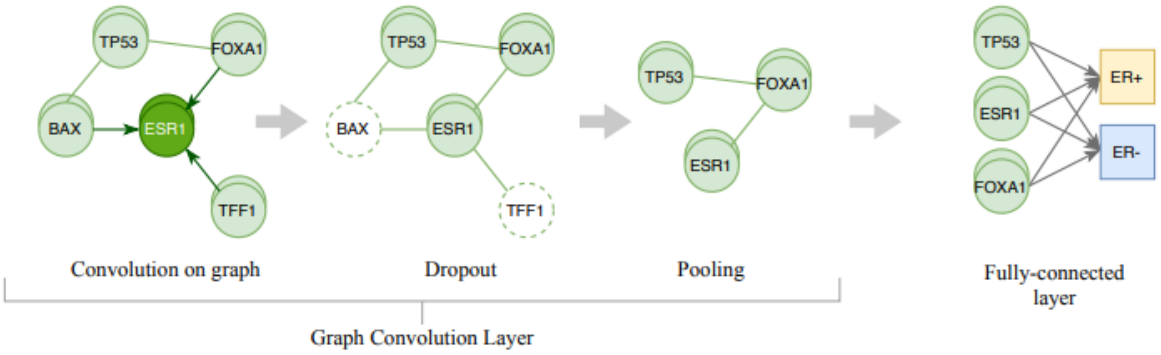

Figure 1: Overview of the OntoGCN applied to gene expression data for patient classification.

- https://arxiv.org/pdf/2011.10998.pdf
- Each patient is a graph instance
    - Node: 3 different ontologies
    - Edges: k nearest genes
- Improved performance over graph-based and non-graph-based baselines on different tasks of cancer subtyping even with low data regimes (100s)

Table 2: Performance comparison of the graphs directing GCN structure on the PAM50 classification task (accuracy ± std). The GCN used expression values as the nodes' input, without the node embedding mechanism.

	PAM 50	
	train size=100	train size=1500
OntoGCN	**72.3±2.9**	**81.2±0.8**
Random graph	68.9±3.9	78.0±4.7
GeneMANIA graph [18]	71.7±2.6	80.1±2.1
STRINGdb graph [26]	71.2±2.5	80.7±1.2
Random Forest	69.3±1.1	78.4±1.0
Multi-Layer Perceptron	60.1±5.0	77.9±2.5

    - Feature selection setting
- Diet approaches: GRL for network parametrization
    -

Readings:
https://web.stanford.edu/class/cs224w/