Computer Science Tripos – Part II – Project Proposal

Graph neural networks for age prediction from neuroimaging data

Kamilė Stankevičiūtė (ks830), Gonville & Caius College

October 17, 2019

Project Originator: Tiago Azevedo

Project Supervisors: Tiago Azevedo, Alexander Campbell, Prof Pietro Liò

Directors of Studies: Dr Timothy M. Jones, Dr Graham Titmus

Project Overseers: Prof Jon Crowcroft, Dr Thomas Sauerwald

Introduction

A graph neural network (GNN) is a type of a neural network that operates on graph inputs and is used for tasks like node classification, link prediction and clustering (geometric deep learning). GNNs have recently become popular and proved successful in a broad range of real-world applications, such as text and image classification, knowledge graphs, and interaction modelling in physical and biological systems. [1]

One domain where graphs offer a natural representation is social networks and *populations*, with nodes representing individuals (their features and labels), and edges corresponding to associations between individuals according to some heuristic or a formally defined similarity metric. The reason why such graph representation is considered to be useful in the geometric deep learning context is that the network can make use of both the individual features (node feature vectors) and the overall trends in the population through pairwise similarities (graph edges), [2] inferring the label of an individual node both from the node itself and from its neighbourhood.

Project description

This project was inspired by Parisot et al.'s [2, 3] state-of-the-art application of a type of a GNN called Graph Convolutional Network (GCN) to the population graphs of healthy controls and patients with neurological or neurodegenerative disorders. In these papers, the GCN (adapted from Kipf and Welling [4]) was used in a semi-supervised manner for two classification tasks: 1) prediction of autism spectrum disorder (ASD) from the ABIDE dataset and 2) prediction of a progressive form of Mild Cognitive Impairment (MCI) that develops into Alzheimer's disease (AD) from the ADNI dataset.

Moreover, a recent paper by Kaufmann et al. [5] has linked the incidence of common brain disorders, including ASD, MCI, and AD as well as others, to the deviation between chronological and biological brain ageing. These results suggest that being able to estimate

the subject's age from the neuroimaging data may be important in understanding the mechanisms of those disorders and helping physicians to find more effective treatments.

The aim of this project will therefore be to adapt the population graph approach of Parisot et al. [2, 3] to a regression task on the UK Biobank dataset, predicting the subject's age based on neuroimaging data, and comparing it to another successful geometric deep learning architecture such as the Graph Attention Network. [6] The performance of the networks will be evaluated on the standard metrics, e.g. the coefficient of determination r^2.

Starting point

The source code for the implementation of Kipf and Welling's [4] GCNs and Parisot et al.'s [2, 3] first classification task is publicly available online.[1,2]

I will be using PyTorch for this project because of its support for machine learning on structured graph data. In particular, PyTorch Geometric (PyG)[3] – a geometric deep learning extension library – will make the implementation, iteration and extensions to the model more flexible in addition to performance improvements and simplified APIs.

I have experience with the basics of TensorFlow[4,5] and no experience with PyTorch or graph neural networks. I have attended or will study (possibly in advance) the CST courses related to the subject of this project, such as IA Machine Learning and Real-World Data, IB Artificial Intelligence, II Data Science, II Bioinformatics, and II Machine Learning and Bayesian Inference.

Dataset

I will be using the data from the UK Biobank, kindly preprocessed and provided by Dr Richard Bethlehem of the Department of Psychiatry.

The UK Biobank is a large dataset containing comprehensive phenotypic, genetic, MRI and other data from the total of over 500,000 participants.[6] In this project, I will be using the subset of this dataset with only those subjects who had the neuroimaging data collected and preprocessed (approximately 20,000 participants). This includes both structural (T1, T2 FLAIR) and functional (resting state fMRI) data, preprocessed with the standard UK Biobank pipelines[7] and additionally denoised and parcellated (in several common parcellations) by Dr Bethlehem.

[1] https://github.com/tkipf/gcn
[2] https://github.com/parisots/population-gcn
[3] https://github.com/rusty1s/pytorch_geometric
[4] Five-course Deep Learning specialisation by deeplearning.ai on Coursera
[5] Google's Machine Learning Crash Course and follow-up courses.
[6] https://www.ukbiobank.ac.uk/participants/
[7] https://biobank.ctsu.ox.ac.uk/crystal/crystal/docs/brain_mri.pdf

Work to be done

The following are lists of explicit deliverables to be implemented.

Graph neural network framework

G1. The data is preprocessed into features and is ready for analysis.

G2. Definition of the similarity metric to be used in connecting the graph. The graph is connected based on that similarity metric to be processed by graph neural networks.

G3. Implementation of Kipf's GCN [4] for the age regression task.

G4. Implementation of the Graph Attention Network for comparing its performance to the Graph Convolutional Network.

Evaluation framework

E1. Comparison of the alternative graph neural network models using the coefficient of determination r^2.

Success criteria

The project will be successful if the following items will have been implemented.

SC1. Representation of the UK Biobank data as a population graph with nodes representing the individuals and edges representing associations between them based on pairwise similarity.

SC2. The Graph Convolutional Network for age regression on the population graph.

SC3. The Graph Attention Network for the same task.

SC4. The evaluation framework for comparing the performance of the two graph neural networks.

Evaluation of the project

The performance of the graph neural networks will be measured across several metrics. The main metric to evaluate a regression task, in contrast the classification in Parisot et al. [3], is the coefficient of determination r^2.

Possible extensions

PE1. An additional metric that could be used to evaluate the performance of the networks is *robustness* to missing or noisy data. Robustness, which could be defined as *the rate at which the predictive power drops as more information is removed from the nodes*, would reveal how important is the neighbourhood (edge) information for accurate predictions compared to the node features only.

PE2. Implement spectral filter computation with *Cayley polynomials* instead of using Chebyshev polynomials. Cayley polynomials have been introduced in a paper by Levie et al. [7] and were mentioned in Parisot et al. [3] as a possible improvement.

PE3. The main implementation of the graph neural network relies on manually handcrafted features from preprocessed brain imaging data. Time permitting, an extension could be to create a package that can be used after any standard neuroimaging preprocessing pipeline (e.g. with results in BIDS[8] format) to extract these features, and possibly improve upon as well as create new ones. This would make execution of the model more efficient, robust and generalisable.

PE4. Implement weighted edges in the Graph Convolutional Network and Graph Attention Network, as the main implementations will have binary edges.

Timetable and milestones

Michaelmas weeks 0–1

- Work on project proposal.

Milestones. Submit Phase 1 report by 14/10/2019. Submit draft proposal by 18/10/2019.

Michaelmas weeks 2–4

- Get access to the UK Biobank data and get familiar with its features.
- Define a possible graph similarity metric.

Milestones. Submit final project proposal by 25/10/2019.

Michaelmas weeks 5–6

- Write code for connecting the nodes based on a similarity metric.
- Connect the nodes (with their features) into a graph.
- Start working on the implementation of the Graph Convolutional Network (e.g. define loss and random label removal for semi-supervised training, start implementing the layers).

Michaelmas weeks 7–8

- Work on the implementation of layers for the Graph Convolutional Network. Compute the general performance metrics.
- Start working on Graph Attention Network implementation for the same task.

Michaelmas vacation

- Continue working on and finish the neural network implementations, compute performance metrics.
- Work on graph neural network evaluation: implement the robustness measurement framework.
- Measure the robustness of the neural networks.
- Start writing the dissertation and the project progress report.

[8]https://bids.neuroimaging.io

Milestones. Complete the implementation of the main part of the project.

Lent weeks 0–2

- Finish the progress report, prepare for the presentation.
- Implement Cayley polynomials.
- Start working on the data preprocessing pipeline.

Milestones. Submit progress report by 31/01/2020.

Lent weeks 3–5

- Continue implementing the data preprocessing pipeline.
- Start working on the implementation of weighted edges.

Lent weeks 6–8

- Finish implementing the data preprocessing pipeline.
- Finish implementing the weighted edges.
- Continue working on the dissertation write-up.

Easter vacation

- Complete the dissertation draft and send it for review.
- Edit the draft based on the feedback received.

Milestones. Send out the complete draft for review by 27/03/2020. Submit dissertation early by 20/04/2020.

Easter weeks 0–2

Time reserved for any unexpected issues.

Resource declaration

For this project I will be using my personal MacBook Pro (2019, with 1.4 GHz Quad-Core Intel Core i5 processor and 8GB of RAM). Training the model will require the use of GPUs provided by the Computational Biology Group (as confirmed by Prof Pietro Liò). To prevent any loss of data, both the source code and the LaTeX source will be stored on my machine, private GitHub repositories, and Google Drive, as well as regularly backed up on an external HDD.

References

[1] Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, et al. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.

[2] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, et al. Spectral graph convolutions on population graphs for disease prediction. *MICCAI*, 2017.

[3] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, et al. Disease prediction using graph convolutional networks: Application to Autism Spectrum Disorder and Alzheimer's disease. *Medical Image Analysis*, 48:117–130, August 2018.

[4] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

[5] Tobias Kaufmann, Dennis van der Meer, Nhat Trung Doan, Emanuel Schwarz, et al. Common brain disorders are associated with heritable patterns of apparent aging of the brain. *Nature Neuroscience*, 22(10):1617–1623, 2019.

[6] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. *International Conference on Learning Representations*, 2018.

[7] Ron Levie, Federico Monti, Xavier Bresson, and Michael M. Bronstein. CayleyNets: Graph convolutional neural networks with complex rational spectral filters. *arXiv preprint arXiv:1705.07664*, 2017.