COMP8221 Assignment 2

Project Option 1: Real-world applications of GNNs

In this project, you will identify a specific use case and demonstrate how GNNs and PyG can be used to solve this problem. We welcome use cases across domains (e.g., biological, social, financial), but we want to see use cases that are motivated by real-world problems, e.g., an application that a company would actually use.

Given a use case, we expect you to:

- **identify an appropriate *public* dataset**,
- **formulate the use case as a clear graph ML problem**,
- **demonstrate how GNNs can be used to solve this problem**.

Note that **it is not enough to simply apply an existing GNN model in the practical workshops to a new dataset.** We expect to see some form of **novelty** (e.g. developing a new GNN method for your specific problem, improving on existing methods) and **comprehensive analyses** (e.g. ablation studies, comparison between multiple model architectures) for the project.

Example graphs & datasets:

Below, we list examples of several real-world application domains, specifying how graphs and tasks are defined in these domains.
You can use datasets available in PyG(https://pytorch-geometric.readthedocs.io/en/latest/modules/datasets.html).
You are free to explore other public datasets, application domains, models, and public datasets yourself.

Recommender systems
Graphs:

- Nodes: Users, items
- Edges: User-item interactions

Tasks:

- Predicting the edge ratings.
- Predicting edge existence.

Fraud detection in transaction graphs
Graphs:

- Nodes: Financial users (customers, banks)
- Edges: Transaction (money and amount sent)

Tasks:

- Edge classification - predict which edges are fraudulent.

Friend recommendation
Graphs: social network

- Nodes: users
- Edges: potentially heterogeneous -- friend, follow, reply to, message, like, etc.

Tasks:

- Recommending/ranking new friends for user.

Author collaboration networks

Graphs:

- Nodes: Authors
- Edges: Author collaboration

Tasks:

- Predicting future author collaboration.

Project Report (30 points)

- *Project Summary (20 points):*
- Minimum 5 page, maximum 15 pages
    - **Section 1: Motivation & explanation of data/task (4 points)**
        - Why did you choose the dataset? Describe your motivation and the dataset and the task (2 points)
        - How did you pre-process the dataset? Including normalization or other necessary techniques. For example, if the dataset is too dense/big and requires more computation power than you have access to, what procedure did you choose to tackle the problem? (2 points)
    - **Section 2: Appropriateness & explanation of model(s) (4 points)**
        - Which graph ML model(s) did you use? Describe your model(s) (Try to include figures and equations for demonstration, **simply apply an existing GNN model implemented in the practical workshops to your chosen dataset is not allowed. Create your own design.** It can be new GNN structure for your specific problem or improving on existing methods. You can also refer to existing research papers/ related blog posts/ tutorials for inspiration.) (2 points)
        - Why do you choose the model(s)? For example, why do you choose GCN as your GNN layer instead of GAT? Why do you choose 3 as your number of GNN layer, etc. (2 points)
    - **Section 3: Insights & results (4 points)**
        - Report primary results of your model(s) in your chosen evaluation metrics (e.g., accuracy, recall, precision, …) (2 points)
        - Include insights or explanations for why your model(s) work well or do not work well for the task (Your design of GNN **does not always work: in such cases, an illustration using experimental results and good discussion of it can still receive full credit.**) (2 points)
    - **Section 4: Comprehensive analysis (4 points)**
        - Report ablation studies of your model(s). (That is, systematically remove some design from your model architecture and see how it will affect the model performance.) (2 points)
        - Comparison between multiple model architectures. (**Only one model in the comparison is required to be your own design**. You are welcome to design multiple models on your own and then compare. You can also compare to existing models in research papers, github posts, etc. **You are required to briefly introduce the models you compared to and why you choose them in section 2. If you choose existing models, you also need to reference them in the report.**) (2 points)
    - **Include figures for illustration (4 points)**

        - Visualization of the graph data. (1 points)
        - Architecture of your proposed model(s) (1 points)
        - Visualization of evaluation metric, for example, learning curves, bar charts to show accuracy, etc. (at least two figures expected for 2 points)

- *Code (10 points):*
    - Code to produce all the results in the project summary:
        - Correctness in data pre-processing part (2 points)
        - Correctness in building model(s) (2 points)
        - Correctness in training and producing results in section 3 (2 points)
        - Correctness in producing comprehensive analysis results in section 4 (2 points)
    - Documentation: class/function descriptions, comments in code (2 points)

Submission Instruction (Due Date: 11:55pm, November 2nd, 2025)

- Submit a project summary that includes the 4 sections and figures as instructed above in pdf format.
- The report should be named as ”2025S2 COMP8221 Assignment 2 your student ID + your name.*”
- Submit your well-documented code and instructions for running it.
- Put your report and your code in one folder named “2025S2 COMP8221 Assignment 2 your student ID + your name” and then compress it for submission.
- Please try your best to use Python >= 3.8 to develop your code to avoid difficulties in grading. Please note, code that uses Python packages outside the standard library are not guaranteed to work.
- You can use either PyTorch or TensorFlow for coding. However, if you prefer to use a different framework, please consult with the course instructor or teaching assistants beforehand.

Project Option 2: Cutting-edge research in DRL

This project helps students to explore Deep Reinforcement Learning in more depth. Novel research ideas are welcome but **are not expected nor required to receive full credit**. In this project, you will identify one research paper in Deep Reinforcement Learning.

In this project, we expect you to:

- **understand a high-quality research paper and how theoretical concepts are implemented practically,**
- **reproduce the results presented in the paper,**
- **build on top of the authors’ code, to implement your own ideas how to further improve the current algorithm.**

Note: **this project is more manageable if you are already comfortable with PyTorch/ Tensorflow and deep learning.** This project can help you to build and strengthen your abilities in doing scientific research. **Some tasks can be challenging but if you are interested in doing research, I would recommend you give it a try.**

Recommended papers:

Below is a list of recommended research papers. If you're looking for additional papers in DRL, it may be useful to look at the proceedings of the top ML conferences, such as NeurIPS, ICML, ICLR and AAAI. **Usually, authors are encouraged to release their code, but it's better to check on your own that the code is available and up-to-date. Sometimes, package versions/installations can be a big headache for reproducibility.**

1. *"Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor"* by Tuomas Haarnoja et al.
2. *"Addressing Function Approximation Error in Actor-Critic Methods"* by Scott Fujimoto et al.
3. *"Distributed Prioritized Experience Replay"* by Dan Horgan et al.
4. *"A Deeper Look at Experience Replay"* by Shangtong Zhang et al.
5. *"Rainbow: Combining Improvements in Deep Reinforcement Learning"* by Matteo Hessel et al.
6. *"Sample-Efficient Deep Reinforcement Learning via Episodic Backward Update"* by Eunhyeok Park et al.
7. *"Maxmin Q-learning: Controlling the Estimation Bias of Q-learning"* by Qingfeng Lan et al.
8. *"Randomized Ensembled Double Q-Learning: Learning Fast Without a Model"* by Xinyue Chen et al.
9. *"Image Augmentation Is All You Need: Regularizing Deep Reinforcement Learning from Pixels"* by Denis Yarats et al.
10. *"A Minimalist Approach to Offline Reinforcement Learning"* by Scott Fujimoto et al.
11. *"Diffusion Policies as an Expressive Policy Class for Offline Reinforcement Learning"* by Zhendong Wang et al.
12. *"Conservative Q-Learning for Offline Reinforcement Learning"* by Aviral Kumar et al.

Project Report (30 points + 2 bonus points):

- *Project Summary (24 points):*
- Minimum 5-page, maximum 15-page limit.

    - **Section 1: Motivation & explanation of research methods (4 points)**
        - Which paper do you intend to work on? Describe the overall problem that the paper is trying to solve, and the specific methods used in the paper. (2 points)
        - Why do you choose this paper? Why are these methods interesting? What is their significance in DRL? (2 points)

    - **Section 2: Explanation of implementation (6 points)**
        - Code snippets explanation: **identify authors' code and link the code with the methods introduced in the paper.** Attach code snippets of the **main methods** proposed in the paper and explain the implementation. **(only the main novelty part in the paper, for example, TD3 proposed Clipped Double Q-learning, but the policy network, replay buffer, etc are the same, then you only need to identify the part related to Clipped Double Q-learning)** For explanation part: what are the code snippets mainly doing? Any tricks that are not mentioned in the paper? Any suggestions to improve or modify the implementations? This is to help you understand **how theoretical concepts are implemented practically**.) (4 points)

        - Create a flowchart to illustrate structure and functionality of the authors' code. (2 points)

    - **Section 3: Reproducibility (8 points)**
        - Reproduce primary experiments in your selected paper and compare the results. (**You do not need to reproduce all the primary experiments in your selected paper**, for example the authors may experiment with a new method that requires more computation power than you have access to, but also present results for a baseline method, in which case you could elect to reproduce only the baseline results. You can also select some environments to reproduce instead of all of them. **Explain what you choose to reproduce and summarize and compare your results with results reported in the paper. Figure illustration is recommended. You can and should use authors' code.**) (4 points)
        - Reproduce the ablation study in your selected paper and compare the results. (**You do not need to reproduce all the ablation studies in your selected paper. Explain what you choose to reproduce and summarize and compare your results with results reported in the paper. You can and should use authors' code if available.** Most research papers should include some ablation studies, however, if authors do not include ablation studies in the paper, design ablation studies on your own and modify authors' code for the experiments.) (4 points)

    - **Section 4: Innovation (6 points)**
        - Build on top of the authors' code, modify the current algorithm to embed at least one other technique introduced in related papers that are not included in the selected paper/ or your own ideas. The goal is to further improve the current algorithm. Explain your modification (what is the other technique), the motivation for the modification (why do you think it may improve the current performance) and report the results. (It's likely your modification **does not work well, and the results may be worse. In such cases, some illustration using experimental results and a discussion is encouraged and can still receive full credit.** The whole goal for this section is to encourage your innovation and simply try your ideas.)

- *Code (6 points + 2 bonus points):*
    - Code built on top of the authors' code:
        - Correctness in modifying authors' code for ablation studies if ablation studies are missing in the selected paper. (bonus 2 points)
        - Correctness in modifying authors' code for experiments in section 4 (4 points)
    - Documentation: class/function descriptions, comments in code (2 points)

Submission Instruction (Due Date: 11:55pm, November 2nd, 2025)

- Submit a project summary that includes the 4 sections as instructed above in pdf format.

- The report should be named as ”2025S2 COMP8221 Assignment 2 your student ID + your name.*”
- Put your report and code in one folder named “2025S2 COMP8221 Assignment 2 your student ID + your name” and then compress it for submission.
- If you directly modified authors’ code by adding your own functions for section 4, please submit the modified version of the authors’ code and instructions for running it. **Place comments at the top of the code to indicate where modifications have been made**.
- If your code for section 4 is in a separate file, please submit it along with the authors’ code and instructions for running it and identify which file is your work.