Data Mining

Homework 4

Due: 19/12/2021, 23:59

Instructions

You must hand in the homeworks electronically and before the due date and time.

The first homework has to be done by each **person individually**.

Handing in: You must hand in the homeworks by the due date and time by an email to Andrea (`mastropietro@diag.uniroma1.it`) that will contain as attachment **(not links to some file-uploading server!)** a .zip file with your answers. The filename of the attachment should be `DM_Homework_1__StudentID_StudentName_StudentLastname.zip`;
for example:
`DM_Homework_1__1235711_Robert_Anthony_De_Niro.zip`.
The email subject should be
`[Data Mining] Homework_1 StudentID StudentName StudentLastname`;
For example:
`[Data Mining] Homework_1 1235711 Robert Anthony De Niro`.
After you submit, you will receive an acknowledgement email that your project has been received and at what date and time. If you have not received an acknowledgement email within 2 days after you submit then contact Andrea.

The solutions for the theoretical exercises must contain your answers either typed up or hand written clearly and scanned.

For any questions on the homework, clarifications, and so on, contact Andrea (`mastropietro@diag.uniroma1.it`).

For information about collaboration, and about being late check the web page.

Problem 1. Graph Learning on Biological Data

It is well known that graphs are everywhere. Complex systems are defined by the relationships among their elements and so they can be represented as networks. Such complex systems cover all scales, from protein–protein interaction networks in cells to social interaction networks among people. For this exercise, you will explore and work with **protein–protein interactions** (PPIs). In this network nodes are genes/proteins that are connected with an edge if a physical interaction exists between them. You can find the Biogird dataset containing protein interactions from here. If this network is too large you can download a reduced version with the first 10K interactions from here.

The task you have to perform is **node classification**. You have to classify a node as **associated to a disease** or not (binary classification). For this exercise you will compare two machine-learning models: the first one is a model of your choice that relies on Node2Vec embeddings as features vectors, the second one is a graph neural network model.

1. **Data Preparation.**

   You will start by preparing your data:

   - **Download data**: Download the provided PPI files and **gene–disease associations**

(GDAs) data from DisGeNET ("ALL gene–disease associations"). You need to register to download.

- **Data processing**: Choose a disease of interest (maybe select a disease with a reasonable number of associated genes) and assign labels to the genes in the PPI network (for instance 1 if the gene is associated and 0 otherwise).

2. **Model training.**

   As previously said, you will compare two models:

   - **Node2Vec embedding**: Use Node2Vec to create node embeddings. Once you have the embedding for any gene you can feed them to a classifier of your choice: MLP, SVM, whatever you like the most :-).
   - **Graph neural network**: instead of building embeddings using Node2Vec, let a GNN learn such embdeddings and perform node classification. Use any GNN you want for this purpose (GCNs are more suitable for semi-supervised learning application, like this one, but feel free to try GATs, GINs, or any other GNN-based model).

   Remember to split your data into train, validation, and test sets. The performances on the test are the ones that matter! Compare the classification performances of the two models in terms of **accuracy**, **recall**, **precision**, and **F1-score**. Which technique was able to better characterize the graph and deliver the highest performances? Note that the answer may not be obvious. You may find that a model outperforms the other on a specific performance metric and viceversa, can you tell why? You have to hand in the code along with a report (about 3–5 pages) in which you describe all the steps made (plots are welcome). In particular, show how you handled the data, describe the models you chose and provide tables with the performance metrics. Plot also a confusion matrix. Feel free to add any comment/observation you think to be relevant.

   Hint: you may need to associate a feature vector to each node during the data preparation. Ideally, you should use biology-related features. Since this is not the goal of this HW, you can use `NetworkX` library to extract simple centrality measures for any node (degree, betweenness, eigenvector, closeness, etc.). Those measures can be a rather good feature vector for our purposes.

3. **Bonus–Explainability.**

   You have used the GNN as a black-box model. Do you want to also explain one of its prediction? Choose one of the explainability methods existing in literature, (GNNExplainer, PGExplainer, SubgraphX, GraphSVX, etc.), and graphically show the explanation subgraph for a node of your choice. If you decide to embark on this bonus exercise and you want more insights, just contact Andrea.