

## COMP396

# Summary and brainstorming of Mutual Information maximization on graphs

## Summary

1. Recall
2. Relevant Papers
3. Supporting Material
4. About the previous models
5. Our model (proposition)
6. Questions/clarifications

## Recall

- During our last meeting, I presented my results/corrections for both QM9 and Zinc datasets prediction tasks with SMILES strings and LSTM.
- You assigned me to :
  - read the paper on DGI and InfoGraph
  - think about how to implement our project with DGI

## Relevant Papers

-LEARNING DEEP REPRESENTATIONS BY MUTUAL INFORMATION ESTIMATION AND MAXIMIZATION-><https://arxiv.org/pdf/1808.06670.pdf>

-DEEP GRAPH INFOMAX: <https://arxiv.org/pdf/1809.10341.pdf>

-INFOGRAPH: UNSUPERVISED AND SEMI-SUPERVISED GRAPH-LEVEL REPRESENTATION LEARNING VIA MUTUAL INFORMATION MAXIMIZATION-><https://arxiv.org/pdf/1908.01000.pdf>

## Supporting Material

-Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results-> <https://arxiv.org/pdf/1703.01780.pdf>

FASTGCN: FAST LEARNING WITH GRAPH CONVOLUTIONAL NETWORKS VIA IMPORTANCE SAMPLING-> <https://arxiv.org/pdf/1801.10247.pdf>

Mutual Information Neural Estimation-><https://arxiv.org/pdf/1801.04062.pdf>

## About InfoGraph

### From InfoGraph:

Objective: Obtain embeddings at the whole graph level for unsupervised and semisupervised learning

InfoGraph maximizes the mutual information between the representations of entire graphs and the representations of substructures of different granularity.

InfoGraph\* maximizes the mutual information between unsupervised graph representations learned by InfoGraph and the representations learned by existing supervised methods.

Based on a student-teacher model strategy where student model is trained on the labeled data and teacher model is trained on unlabeled data with InfoGraph. (different from knowledge distillation?)

Knowledge distillation: where a large (teacher) pre-trained network is used to train a smaller (student) network.

Main Graph convolution encoder used: GIN (Graph Isomorphism Network).

Infograph\* loss function combination of a supervised and unsupervised objective function.

## About DGI

Objective: Focuses on learning better patch/node representations that have an overall better summary of the graph they belong to for node classification tasks.

Main Graph convolution encoder used: GCN Kipf & Welling (2016)

## Corruption function alternatives

From DGI:

NB: Accuracy increases the less the Adjacency matrix is modified.

Corruption functions that preserve sparsity perform the best.

- $\neg A=A$  but corrupts the features,  $\neg X$ , via row-wise shuffling of  $X$
- Isomorphic Adjacency matrix perturbation where we shuffle both  $X$  and  $A \rightarrow \neg A!=A, \neg X!=X$

From InfoGraph:

Batch-wise generation of negative samples  $\rightarrow$  Involves using (Global, patch representations) from samples that do not match in order to get negative examples.

## Objective functions (general for DIM)

Noise Contrastive estimator:

Pros : Generally best results and works for graphs

Cons: Requires many negative samples

Jensen- Shannon estimator

Pros : Stable

Cons: undesirable statistical properties

## Materials & Resources

InfoGraph Code: <https://github.com/fanyun-sun/InfoGraph>

Deep Graph Infomax (Original): <https://github.com/PetarV-/DGI>

Deep Graph Infomax:

[https://github.com/rusty1s/pytorch\\_geometric/blob/master/torch\\_geometric/nn/models/deep\\_graph\\_infomax.py](https://github.com/rusty1s/pytorch_geometric/blob/master/torch_geometric/nn/models/deep_graph_infomax.py)

## Our Model

Objective: Mutual information maximization between encoded SMILES strings feature map and graph level representation feature map or summary vectors?

Goal: To be able to differentiate matching pairs of graph level representations and encoded SMILES Strings representations?

Semi-supervised setting: what would it mean in this setting? Use of inductive GNN with a supervised encoding of SMILES?

Reasoning

### **Graph representations:**

Pros

- Graph neural networks have the most accurate results for machine learning molecular property prediction tasks.

- Natural representation of molecules

Cons: Very sophisticated 3D models, initialization is similar to handcrafted features

- Time complexity can be a bit high depending on the model

### **SMILES strings:**

Pros: -One of the simplest model to encode

- Faster and smaller time and space complexity respectively during training
- High availability in multiple databases ChEMdb

Cons: Accuracy level usually lower than SOTA graph results (From my experiments and observations)

Combining both models would possibly best address the engineering shortcomings of previous models while achieving the best accuracy.

Both models are invariant to graph Isomorphism and canonicalized smiles strings are computed in a DFS order.

## Proposition

Smiles +LSTM + graph nodes (ex k-gnn? )

- 1)Local global - Noise contrastive approach- Noise contrastive estimator
  - Corruption function based on row-wise shuffling of graph
- 2)- Jensen Shannon divergence- Jensen Shannon estimator
  - Negative batch sampling (permutate information)

Global Global

- 3)- Noise contrastive approach- Noise contrastive estimator
  - Corruption function based on row-wise shuffling of graph
- 4)- Jensen Shannon divergence- Jensen Shannon divergence
  - Negative batch sampling (permutate information)

Unsupervised :


I'm not sure if this could work

LSTM trained supervised + unsupervised graph ()

- 1)Local global - Noise contrastive approach- Noise contrastive estimator
  - Corruption function based on row-wise shuffling of graph

## Questions:

1. Which graph do we want to use?
2. What's the most expressive graph?

- 
3. What is our end goal? Achieve a matching or get a better representation. (if not both)  
What information to keep on features (charges, hybridization, and distance between nodes are not mutual )?
  4. Are we interested in knowledge transfer?
  5. Clarify inductive DGI?
  6. Which representation will be considered as global/local? Global infomax as in DIM instead?
  7. What corruption function to use and on which representation to use it?

