

We present GEVI (Graph Embedding Visual Inspector), an evaluation framework for graph embedding. The rationale behind our framework is to provide developers, end users and researchers with easy-to-use interfaces that allow for the agile, fine-grained and uniform evaluation of graph embedding approaches on multiple and real datasets.

How to setup GEVI :

1.1. Downloading and running the system:

1.1.1. Prerequisites :

GEM is tested to work on Python 3.0

The required dependencies are

Numpy \geq 1.12.0

SciPy \geq 0.19.0

Networkx \geq 1.11

Scikit-learn \geq 0.18.1

Theano \geq 0.9.0

Keras = 2.0.2

1.1.2. Install :

The easiest way to install GEVI to your local machine is to download the source code from GitHub using Git or OS command line.

```
Windows PowerShell
Copyright (C) 2015 Microsoft Corporation. Tous droits réservés.

~\Documents\GitHub> git clone https://github.com/PFE-Passau/Evaluation_Framework_For_Graph_Embeddings.git
```

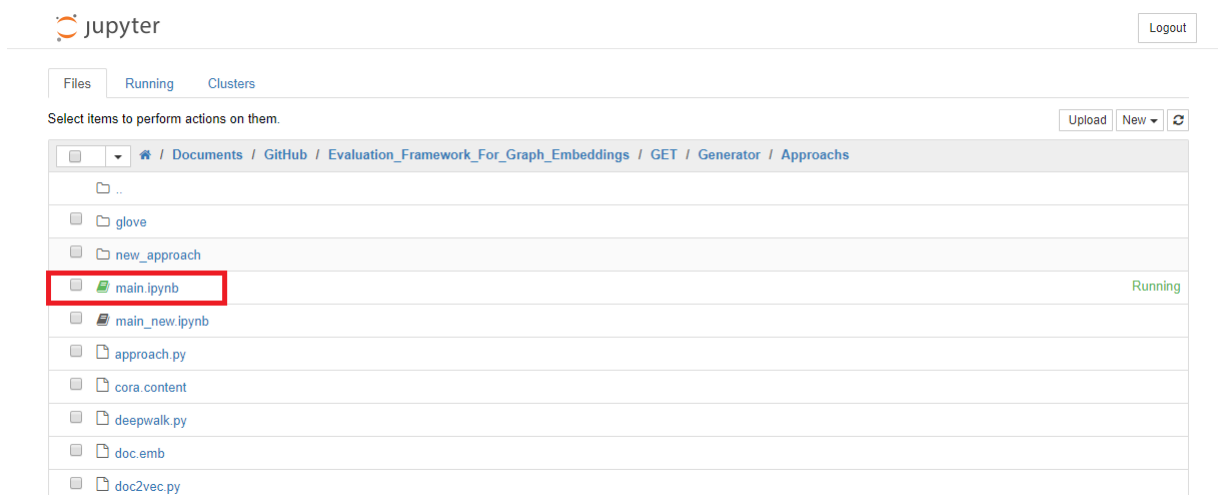
1.2. Repository Structure:

- GET/Generator/Approaches: 7 Existing approaches for graph embedding in python language scripts.
- GET/Generator/Datasets: 10 Existing datasets with different formats: edgelist, label and text.
- GET/Evaluator/:4 Existing tasks for graph embedding: link prediction, node classification, graph visualization and word embedding visualization in python language scripts.

2. How to generate graph embedding using existing approach and existing dataset in GEVI?

1- In your "GET/Generator/Approachs" directory, you can execute GEVI's main script using:

(i) Jupyter notebook by clicking on main.ipynb



(ii) OS command line (terminal)

```
Invite de commandes - python main.py
Microsoft Windows [version 10.0.10240]
(c) 2015 Microsoft Corporation. Tous droits réservés.

C:\Users\asus>cd Documents
C:\Users\asus\Documents>cd GitHub
C:\Users\asus\Documents\GitHub>cd Evaluation_Framework_For_Graph_Embeddings
C:\Users\asus\Documents\GitHub\Evaluation_Framework_For_Graph_Embeddings>cd GET
C:\Users\asus\Documents\GitHub\Evaluation_Framework_For_Graph_Embeddings\GET>cd Generator
C:\Users\asus\Documents\GitHub\Evaluation_Framework_For_Graph_Embeddings\GET\Generator>cd Approachs
C:\Users\asus\Documents\GitHub\Evaluation_Framework_For_Graph_Embeddings\GET\Generator\Approachs>python main.py
Select the type of the dataset (Edgelist, Label or Text information):
```

2- GEVI will ask you firstly for the format of your dataset to test it (Edgelist , Label or text information).

- ✓ If you write “edgelist”, GEVI will offer you 3 graph embedding approaches which correspond to the specific format such as Node2vec, LINE or DeepWalk.

```
Select the type of the dataset (Edgelist, Label or Text information):edgelist
According to your Dataset characteristics, you must select from these graph embeddings approaches
001) Node2vec
002) DeepWalk
003) LINE
Enter option: |
```

- ✓ If you write “label”, GEVI will offer you 2 graph embedding approaches which correspond to the specific format such as Doc2vec and Paper2vec.

```
Select the type of the dataset (Edgelist, Label or Text information):label
According to your Dataset characteristics, you must select from these graph embeddings approaches
004) Doc2vec
005) Paper2vec
Enter option: |
```

- ✓ If you write “text”, GEVI will offer you 2 graph embedding approaches which correspond to the specific format such as Word2vec or GloVe.
- ✓

```
Select the type of the dataset (Edgelist, Label or Text information):text
According to your Dataset characteristics, you must select from these graph embeddings approaches
006) Word2vec
007) Glove
Enter option: |
```

3- Then, you have to take in the name of your dataset among the existing datasets like Facebook , Twitter or Cora etc.

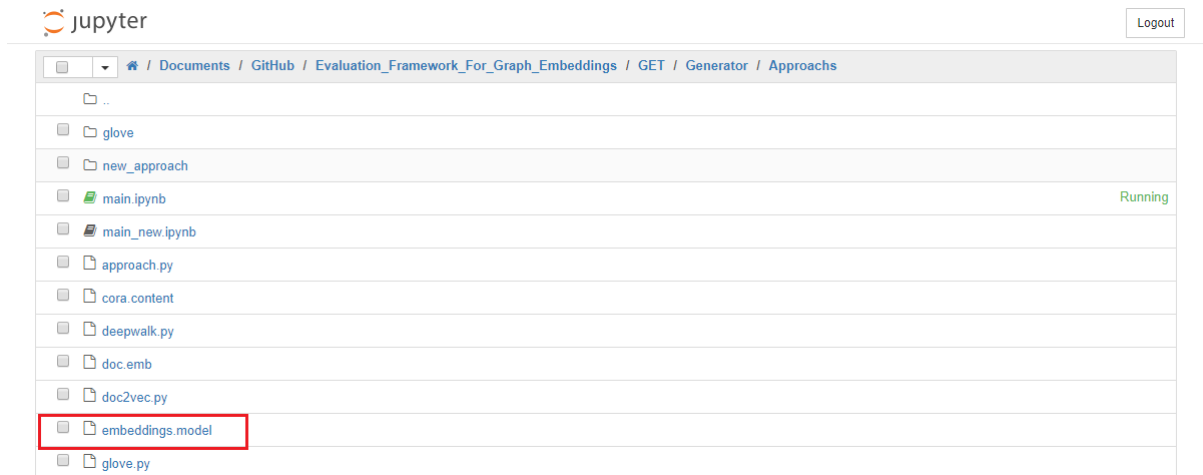
```
Select the type of the dataset (Edgelist, Label or Text information):edgelist
According to your Dataset characteristics, you must select from these graph embeddings approaches
001) Node2vec
002) DeepWalk
003) LINE
Enter option: 1
Using TensorFlow backend.
Enter graph edgelist filename: facebook.edgelist |
```

4- Furthermore, GEVI will recommend you to choose the best bag of hyperparameters for your model thanks to recent researches which works on graph embedding approaches at the moment while our approaches frequently require careful tuning of model hyperparameters, regularization terms, and optimization parameters but in the future GEVI will adopt new concept of optimization in which a learning algorithm's generalization performance is modeled as a sample from a Gaussian process.

Based on previous experiments the best in-out and return hyperparameters are {0.25, 0.50, 1, 2, 4}

Enter in-out parameter:

5- Finally, you will give the name of the new text file which will contain the embedding and you will get it in the current folder.



3. How to generate embedding using new approach or new dataset in GEVI?

3.1. How to add a new Dataset?

At the moment, there is only one possibility to add a new dataset to GEVI:

-Transforming your dataset into a txt, csv or xls file and put it into "GET/Generator/Datasets" folder.

3.2. How to add a new Approach?

For adding a new approach to GEVI, you should follow these steps:

- 1- Create an empty python script in "GET/Generator/Approaches/new-approach".
- 2- The script must contain in its skeleton class named new which contains 2 functions:
 - * Get-information function which return the format of dataset in string type that we can apply the new approach on it (edgelist , text or label)
 - * run function which contains the new approach's code in python language.

```
class new():
    def get_information():
        return "text"
    def run():
        print("code")
```

3.3. How to add a new Task?

For adding a new task to GEVI, you should adjust the new task with the embedding characteristics and create a python script on "GET/Evaluator" directory.

4. How to evaluate graph embedding approach in multiple tasks?

1- In your "GET/Evaluator" directory, you can execute GEVI's main using:

(i) Jupyter notebook by clicking on main.ipynb .

(ii) OS command line (terminal).

2- Select among the 4 tasks: Node Classification, Link prediction, Graph Visualization and word embedding visualization.

```
Select the specific task
1) Node Classification
2) Link Prediction
3) Word embeddings Visualizaion
4) Graph Visualizaion
Enter option: 1
```

```
C:\Users\asus\Anaconda3\lib\site-packages\gensim\utils.py:860: UserWarning: detected Windows; aliasing chunkize to chunkize_serial
warnings.warn("detected Windows; aliasing chunkize to chunkize_serial")
Using TensorFlow backend.
```

3- Take in the embedding file name which must be in your "GET/Evaluator" directory.

4- GEVI will generate text file named evaluation with contains the performance metrics for your specific task.