Οδηγίες εγκατάστασης (Windows)

==========

Εγκατάσταση neo4j community edition 4.0.0

==========

1. Εγκατάσταση Java Development Kit (jdk) 11
   (μεταγενέστερες εκδόσεις ενδέχεται να μην είναι συμβατές)
   και προσθήκη του path στα system variables πχ:
   JAVA_HOME : C:\Program Files\Java\jdk-11.0.11
   JRE_HOME : C:\Program Files (x86)\Java\jre1.8.0_91

2. Κατέβασμα του neo4j από
   https://drive.google.com/file/d/1tZwJasXVJvlXWMn5mo3SsgwplOaruyYi/view?usp=sharing
   και unzip στον φάκελο C:\

   Προσθήκη στην μεταβλητή Path (System variables)
   C:\neo4j\bin

Τα βήματα 3 έως 5 έχουν εκτελεστεί στο παρεχόμενο neo4j

3. Αλλαγές στο C:\neo4j\conf\neo4j.conf
   Γρ 9 : Αφαίρεση του # ώστε να μείνει κάτι σαν:
   dbms.default_database=neo4j
   Γρ 36, 37 Αλλαγή heap size :
   dbms.memory.heap.initial_size=700m
   dbms.memory.heap.max_size=1024m

4. Προσθήκη plugin apoc

   Κατέβασμα του αρχείου:
   https://github.com/neo4j-contrib/neo4j-apoc-procedures/releases/download/4.0.0.16/apoc-4.0.0.16-all.jar
   στον φάκελο C:\neo4j\plugins

   Δημιουργία αρχείου C:\neo4j\conf\apoc.conf
   Γρ 1 : apoc.import.file.enabled=true

5. Σφάλμα κατά την εκκίνηση:

   Import-Module : The specified module '\Neo4j-Management.psd1' was not loaded because no valid module file was found in any module directory.
   At C:\neo4j\bin\neo4j.ps1:27 char:14

   Αντιμετώπιση : Στο αρχείο C:\neo4j\bin\neo4j.ps1 γραμμή 27
   Αντικατάσταση του Import-Module "$PSScriptRoot\Neo4j-Management.psd1"
   με Import-Module "C:\neo4j\bin\Neo4j-Management.psd1"

---

6. Εκκίνηση neo4j την πρώτη φορά
   i. Στον φάκελο C:\neo4j\bin εκκίνηση cmd
   ii. Εκτέλεση neo4j console
   iii. Μετάβαση στο http://localhost:7474/browser/ μέσω browser κατόπιν εμφάνισης σχετικής οδηγίας.
   iv. Ορισμός username και password που πρέπει να συμφωνούν με το αρχείο **config.py** (default username=neo4j, password=290197)

==========

Εγκατάσταση απαραίτητων βιβλιοθηκών

==========

Με την δημιουργία ενός conda environment απαιτείται εγκατάσταση των παρακάτω:

```
conda install python==3.7.12
conda install pandas==1.3.5
conda install inflect==5.5.2
conda install matplotlib==3.5.1
conda install scikit-learn==1.0.2
conda install distance==0.1.3
conda install python-levenshtein==0.12.2
conda install thefuzz==0.19.0
conda install requests==2.27.1
conda install tqdm==4.64.0
conda install psutil==5.9.0
pip install neo4j==4.4.1
conda install neo4j-python-driver==4.4.2
conda install patool==1.12
conda install seaborn

# Παρακάτω Pytorch για cpu only. Υπάρχει δυνατότητα εκτέλεσης
# εκπαίδευσης του GNN σε GPU ανάλογα με το σύστημα.
# Αντικατάσταση των παρακάτω με την αντίστοιχη επιλογή από :
# https://pytorch.org/get-started/locally/
# https://pytorch-geometric.readthedocs.io/en/latest/notes/installation.html

# pytorch version 1.11
conda install pytorch torchvision torchaudio cpuonly -c pytorch
# pytorch geometric for pytorch 1.11
conda install pyg -c pyg

# BERT
pip install transformers==4.18.0

# torchmetrics version 0.9.1
conda config --add channels conda-forge
conda config --set channel_priority strict
conda install torchmetrics

pip install class-resolver
```

==========

Συνοδευτικά αρχεία

==========

Αρχεία που παρέχονται και οδηγίες αποθήκευσής τους σε περίπτωση που χρειαστεί χειροκίνητη παρέμβαση.

Μέσω Google Drive:

- Neo4j όπως χρησιμοποιήθηκε κατά την υλοποίηση
  https://drive.google.com/file/d/1tZwJasXVJvlXWMn5mo3SsgwplOaruyYi/view?usp=sharing
  Unzip στο φάκελο C:\ όπως ορίζεται παραπάνω

Μέσω GitHub:
https://github.com/ChristinaK97/DDI_thesis_files

- data\ddi.rar
  Τα αρχεία xml του συνόλου δεδομένων
  Unzip στο PROJECT_PATH\data
- data\synonyms_data αρχεία για την λειτουργία εύρεσης συνωνύμων φαρμάκων

==========

Ρύθμιση παραμέτρων

==========

Στο αρχείο του project **config.py** δίνεται η δυνατότητα προσδιορισμού παραμέτρων από το χρήστη όπως το path του project (PROJECT_PATH) και στοιχεία πρόσβασης στο neo4j. Επίσης μέσω του ίδιου αρχείου επιλέγεται η αρχιτεκτονική του μοντέλου, ρυθμίζονται υπερπαράμετροι κ.α.. Σχολιασμός και default τιμές παρέχονται εκεί για κάθε μεταβλητή.
Κατ' ελάχιστο **απαιτείται** ο ορισμός του PROJECT_PATH και ο έλεγχος συμφωνίας των στοιχείων του neo4j.

==========

Εκτέλεση

==========

Εκκίνηση cmd στο root φάκελο (PROJECT_PATH)
Ενεργοποίηση του conda environment που δημιουργήθηκε
call conda.bat activate *env-name*

Κλήση ενός από τα παρακάτω αρχεία:

- Εκτέλεση όλο του pipeline μέσω του **main.py**
- Αφού έχει εκτελεστεί προηγουμένως το main.py για δημιουργία της βάσης, μπορεί να εκτελεστεί μόνο η λειτουργία TRAINING ή INFERENCE του μοντέλου μέσω του **run_only_cls_model.py**

==========

Εκπαίδευση μοντέλων μέσω notebook

==========

Η εκτέλεση της εκπαίδευσης των μοντέλων GNN μπορεί εναλλακτικά να πραγματοποιηθεί μέσω του αρχείου **node_classifier.ipynb**. Απαιτείται πρώτα η εκτέλεση τοπικά για την παραγωγή των dataset. Αυτό μπορεί να γίνει μέσω του **prepair_graph_dataset.py**.
Στην συνέχεια ο φάκελος PROJECT_PATH\data\graph_dataset μπορεί να αποθηκευτεί λ.χ. σε Google Drive μαζί με το notebook για εκτέλεση του βήματος εκπαίδευσης μέσω Colab
(default path: '/content/gdrive/My Drive/Colab Notebooks/graph_dataset')

==========

Εκπαιδευμένα μοντέλα

==========

Διατίθενται έτοιμα τα τρία βέλτιστα μοντέλα:

- M2.3 : BioBERT PubMed base → MLP_{pre} → GIN → MLP_{post}
  https://drive.google.com/file/d/1fR6oOOsVE5d7eoMnbeeCBRX-c_GLDPn4/view?usp=sharing
- M10.2 : SciBERT PAR → GIN → MLP_{post}
  https://drive.google.com/file/d/1CN8P0pxKwWv5nK1uxTKX4Ov7abx_SZE_/view?usp=sharing
- M2.2 : BioBERT PubMed base → GIN → MLP_{post}
  https://drive.google.com/file/d/18mKD6EqKQBWpBwvJzTFthwHbnYCO21RC/view?usp=sharing

Στον φάκελο με όνομα που περιγράφει το επιθυμητό μοντέλο δίνονται:

- data\graph_dataset : Περιέχει το dataset (HeteroData) που προέκυψε από την εφαρμογή του επιλεγμένου BERT
- data\models\classification_model.bin : Το εκπαιδευμένο μοντέλο (MODEL_FILE)
- data\models\classification_model_config.json : Οι παράμετροι του μοντέλου (MODEL_CONFIG_FILE)

Το path συμφωνεί με τη δομή του project, δηλαδή ο φάκελος data πρέπει να αποθηκευτεί στον κεντρικό φάκελο του project PROJECT_PATH.

ΕΝΑΛΛΑΚΤΙΚΑ, για εκτέλεση μέσω notebook (πχ Colab):

1. Ο φάκελος του dataset “graph_dataset” μπορεί να αποθηκευτεί πχ σε Google Drive
   (default path: '/content/gdrive/My Drive/Colab Notebooks/graph_dataset')
2. Τα αρχεία του μοντέλου να ανέβουν στο Colab ή να είναι στον ίδιο φάκελο με το notebook
3. Να εκτελεστεί το τελευταίο κελί που αναφέρεται σε INFERENCE του μοντέλου:
   TrainClassificationModel(mode = INFERENCE)

Οι default ρυθμίσεις path μπορούν να αλλάξουν από το δεύτερο κελί του notebook.

==========

Πρόβλημα autograd σε Windows 7

==========

Κατά την εκπαίδευση του μοντέλου παρατηρήθηκε ότι η εκτέλεση υπολογισμού παραγώγων με το βήμα loss.backwards() οδηγεί σε αδυναμία τερματισμού του προγράμματος. Η εκπαίδευση ολοκληρώνεται και το παραγόμενο μοντέλο αποθηκεύεται, ωστόσο το python.exe συνεχίζει να τρέχει.
Το πρόβλημα παρατηρήθηκε σε σύστημα cpu only με Windows 7 για την έκδοση PyTorch 1.11 και δεν είναι σαφές αν θα παρουσιαστεί σε άλλα συστήματα.
Περισσότερες πληροφορίες:
https://github.com/pytorch/pytorch/issues/29383