ΕΡΓΑΣΙΑ ΓΙΑ ΤΟ ΜΑΘΗΜΑ ΘΕΩΡΙΑ ΔΙΚΤΥΩΝ - ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2022-23

Θωμάς Κυριάκος Πραβινός

ΑΕΜ: 9937

Email: tpravinos@ece.auth.gr

Για την υλοποίηση της εργασίας χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python και οι εξής βιβλιοθήκες :

- NetworkX
- Pandas
- Sklearn
- Community Louvain
- SentenceTransformers
- Numpy

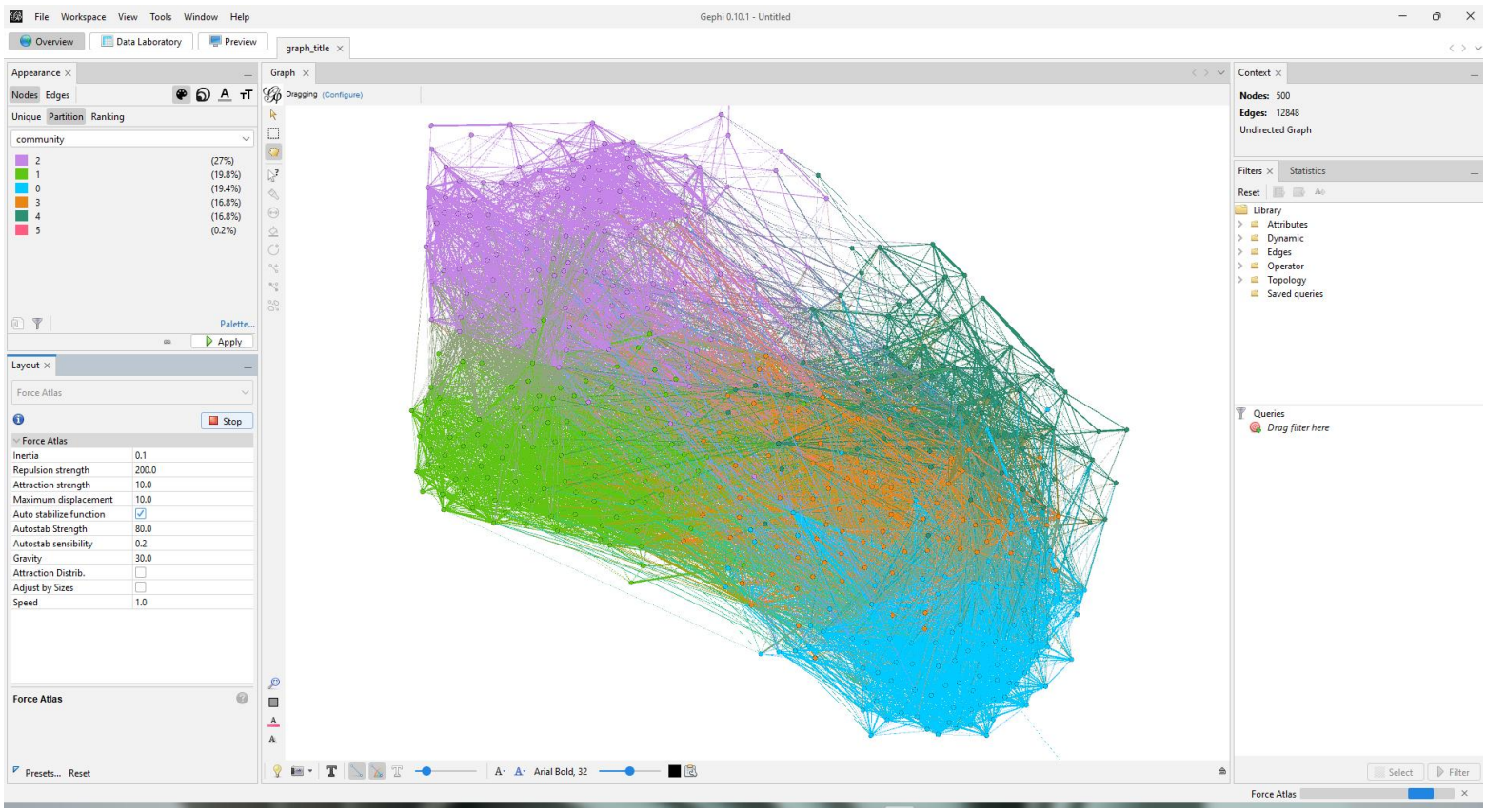
Γενικά, η πορεία που ακολούθησα ξεκίνησε με την φόρτωση του train.csv και την επιλογή των πρώτων 500 στοιχείων του, αφού φόρτωσα τις απαραίτητες βιβλιοθήκες και modules αυτών . Σε πρώτη φάση χρησιμοποίησα το μοντέλο all-MiniLM-L6-v2 ως sentence transformer για την παραγωγή embeddings για τον τίτλο κάθε άρθρου . Στη συνέχεια δημιούργησα έναν γράφο ο οποίος είχε ως κορυφές, τα άρθρα του dataset, και για την δημιουργία των ακμών υπολόγισα το cosine similarity μεταξύ των embeddings των κορυφών ανά δύο, με συνθήκη αν το similarity έχει τιμή μεγαλύτερη του 0,2 τότε προστίθεται μια ακμή μεταξύ των δύο κορυφών . Το όριο το έθεσα στο 0,2 διότι ήθελα στο επόμενο βήμα όπου με τον αλγόριθμο του **Louvain** βρίσκω κοινότητες με το community_louvain.best_partirion, να παίρνω εν τελεί περίπου τον ίδιο αριθμό κοινοτήτων όσο και ο αριθμός των κατηγοριών στο dataset, ώστε να μπορέσω να μπω σε διαδικασία σύγκρισης (αυτό είχε ως αποτέλεσμα να έχω ένα πολύ πυκνό δίκτυο) . Όσα αναφέρθηκαν μέχρι και την συνθήκη για τις ακμές του γράφου πραγματοποιούνται στο πρώτο κελί κώδικα. Στο δεύτερο κελί βρίσκω τις κοινότητες με την χρήση του community_Louvain και μετά υπολογίζω τον αριθμό των κορυφών σε κάθε κοινότητα που προέκυψε αλλά και τα ID των κορυφών αυτών (Υπολογίζεται και το modularity). Στο τρίτο κελί χρησιμοποιώ όλα τα δεδομένα που έχω συγκεντρώσει ώστε να υπολογίσω το **Fowlkes-Mallows Score** μεταξύ των κοινοτήτων που δημιουργήθηκαν και των πραγματικών κατηγοριών. Πρώτα, προσθέτω μια στήλη με όνομα ‘Category’ στο αρχικό dataset και αντιστοιχώ κάθε άρθρο με την πραγματική του κατηγορία. Μετά κάνω group τα άρθρα ανά κατηγορία και δημιουργώ μια λίστα με τα άρθρα σε κάθε κατηγορία. Έπειτα, δημιουργώ 2 λίστες, μια για τις πραγματικές κατηγορίες και μια για τις κοινότητες Louvain οι οποίες έχουν στη θέση κάθε στοιχείου την κατηγορία ή την κοινότητα αντίστοιχα που ανήκει το στοιχείο. Τέλος

υπολογίζω το **Fowlkes-Mallows Score** για αυτές τις δύο λίστες. Ως **Fowlkes-Mallows Score** ορίζεται ο γεωμετρικός μέσος όρος της ακρίβειας και της ανάκλησης ανά ζεύγη μεταξύ δύο ομαδοποιήσεων. Γενικά είναι ένα μέτρο σύγκρισης της ομοιότητας δύο ομαδοποιήσεων, το οποίο παίρνει τιμές από 0 έως 1, με την τιμή 1 να υποδεικνύει την τέλεια ταύτιση. Κάνοντας όλα τα παραπάνω χρησιμοποιώντας τον **Τίτλο** κάθε άρθρου σαν sentence transformer embedding παίρνουμε ένα score 0,5090 ή **50,90%** για τα 500 πρώτα άρθρα του dataset. Στο τέταρτο κελί όλη η διαδικασία επαναλαμβάνεται αλλά αυτή τη φορά χρησιμοποιώ την **Περίληψη** των άρθρων ως sentence transformer embedding και παίρνω τελικά score 0,5638 ή **56,38%** για τα πρώτα 500 άρθρα. Το αποτέλεσμα αυτό ήταν αναμενόμενο καθώς από την περίληψη ενός άρθρου μπορεί να γίνει καλύτερη κατανομή σε κοινότητες από ότι με τον τίτλο, αφού έχει μεγαλύτερο πλήθος πληροφοριών. Σε δοκιμές για μεγαλύτερο αριθμό άρθρων τα score μειώθηκαν ελάχιστα, οπότε κράτησα τα 500 ως ένα δείγμα για εξοικονόμηση χρόνου. Ενδεικτικά για 1000 άρθρα είχα πλέον ποσοστά 49.99 % για τον τίτλο και 55.77 % για την περίληψη , και για να τρέξουν όλα τα κελιά χρειάστηκαν συνολικά περίπου 7,5 λεπτά στον υπολογιστή μου. (Για τα 500 χρειάστηκε περίπου 2 με 3 λεπτά) .

Χρησιμοποίησα το OpenAi ChatGPT ως βοηθό για την σύνταξη του κώδικα και την κατανόηση κάποιων εννοιών αλλά και διάφορες άλλες πηγές στο διαδίκτυο για την ολοκλήρωση της εργασίας . Σαν εναλλακτική, έκανα και μια δοκιμή με το Adjusted rand score αντί του Fowlkes-Mallows score που μοιάζουν αρκετά αλλά τα αποτελέσματα ήταν χειρότερα οπότε κράτησα την υλοποίηση που ανέλυσα παραπάνω.

Απεικόνιση Γράφων με Gephi :

1) graph_title.gexf



2) graph_abstract.gexf