|  |  |
| --- | --- |
|  | **ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ**  **ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ**  **ΚΥΚΛΟΣ ΣΠΟΥΔΩΝ/ΚΑΤΕΥΘΥΝΣΗ: ΥΠΟΛΟΓΙΣΤΩΝ**  **ΤΜΗΜΑ ΕΠΙΒΛΕΠΟΝΤΟΣ: CEID** |

**ΤΙΤΛΟΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ**

*«Πρόβλεψη επιρροής, με χρήση embeddings, σε κοινωνικά δίκτυα και γράφους»*

**Δ Ι Π Λ Ω Μ Α Τ Ι Κ Η Ε Ρ Γ Α Σ Ι Α**

***ΓΚΑΝΑΣ ΙΩΑΝΝΗΣ ΑΜ: 1053577***

***ΕΠΙΒΛΕΠΩΝ: ΜΕΓΑΛΟΟΙΚΟΝΟΜΟΥ ΒΑΣΙΛΙΕΟΣ***

**ΠΑΤΡΑ - 2022**

Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών.

Γκανάς Ιωάννης

© 2022 – Με την επιφύλαξη παντός δικαιώματος

Το σύνολο της εργασίας αποτελεί πρωτότυπο έργο, παραχθέν από τον Γκανά Ιωάννη, και δεν παραβιάζει δικαιώματα τρίτων καθ’ οιονδήποτε τρόπο. Αν η εργασία περιέχει υλικό, το οποίο δεν έχει παραχθεί από τον ίδιο, αυτό είναι ευδιάκριτο και αναφέρεται ρητώς εντός του κειμένου της εργασίας ως προϊόν εργασίας τρίτου, σημειώνοντας με παρομοίως σαφή τρόπο τα στοιχεία ταυτοποίησής του, ενώ παράλληλα βεβαιώνει πως στην περίπτωση χρήσης αυτούσιων γραφικών αναπαραστάσεων, εικόνων, γραφημάτων κ.λπ., έχει λάβει τη χωρίς περιορισμούς άδεια του κατόχου των πνευματικών δικαιωμάτων για την συμπερίληψη και επακόλουθη δημοσίευση του υλικού αυτού.

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω, τον καθηγητή κ. Μεγαλοοικονόμου Βασίλειο για τις γνώσεις που με βοήθησε να αποκομίσω μέσω της υλοποίησης της παρούσας εργασίας. Παράλληλα θα ήθελα να ευχαριστήσω την διδακτορική φοιτήτρια Σμάνη Ελίνα για την καθοδήγηση που μου προσέφερε και την άμεση απάντηση της σε κάθε απορία που προέκυπτε. Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη που μου προσέφερε κατά την διάρκεια της φοίτησης μου.

# Περίληψη

Με τον όρο κοινωνική επιρροή, αναφερόμαστε στο φαινόμενο κατά το οποίο άτομα αλλάζουν την συμπεριφορά τους, έτσι ώστε να προσαρμοστούν στις απαιτήσεις και τις ανάγκες του κοινωνικού περιβάλλοντος στο οποίο είναι ενταγμένα. Παράλληλα, την συναντάμε σε διάφορες εκφάνσεις, όχι μόνο στην καθημερινή πραγματική ζωή, αλλά και στον εικονικό χώρο του διαδικτύου, με το αντίκτυπο της να φαίνεται σε πολλούς τομείς όπως η διαφήμιση, οι εκλογές κ.α. Τα κοινωνικά δίκτυα είναι σύνολα που αποτελούνται από χρήστες οι οποίοι αλληλοεπιδρούν μεταξύ τους, αναπτύσσοντας έτσι σχέσεις επιρροής. Για να χρησιμοποιηθεί αυτή η επιρροή, στον εκάστοτε τομέα, είναι απαραίτητος ο εντοπισμός των χρηστών οι οποίοι είναι οι πιο επιδραστικοί στο κάθε κοινωνικό δίκτυο. Στην παρούσα διπλωματική εργασία, το πρόβλημα πρόβλεψης της επιρροής του κάθε χρήστη αντιμετωπίζεται ως πρόβλημα ταξινόμησης, ενώ η πρόβλεψη της επιρροής των χρηστών επιτυγχάνεται συνδυάζοντας μεθόδους μηχανικής μάθησης με αλγορίθμους δημιουργίας node embeddings. Τα node embeddings είναι διανύσματα με τα οποία γίνεται αναπαράσταση των χρηστών ενός κοινωνικού δικτύου. Για την αξιολόγηση της επιρροής κάθε χρήστη, χρησιμοποιήθηκε το μοντέλο επιδημίας SI με στόχο την δημιουργία προσομοιώσεων διάδοσης ιδεών-απόψεων. Εφαρμόστηκαν έξι αλγόριθμοι μηχανικής μάθησης και τέσσερις αλγόριθμοι δημιουργίας node embeddings. Τα αποτελέσματα που προέκυψαν, από τρία σύνολα δεδομένων πραγματικού κόσμου, διαφέρουν από μέθοδο σε μέθοδο με κάποια να κρίνονται εξαιρετικά και άλλα να είναι πιο χαμηλά.

**Λέξεις κλειδιά**: κοινωνικό δίκτυο, χρήστες με μεγάλη επιρροή, μηχανική μάθηση, ταξινόμηση, μοντέλο επιδημίας SI, node embeddings

# Abstract

Using the term of social influence, we refer to a phenomenon according to which, people change their behavior, to adjust to the requirements of their social environment. In parallel, social influence exists in many forms, not only in everyday real life but in virtual space of Internet and it has impact in many sectors like marketing, elections etc. Social networks are sets of users who interact and influence each other. To use this influence, in each sector, it is necessary to find which users are more influential than others. In this diploma thesis, the problem of predicting each node’s influence is considered as a classification problem and to deal with it we combine machine learning methods and node embeddings. Node embeddings are vectors, that can represent the users of a social network. In addition, for evaluating the influence of each user, we used SI epidemic model with a view to create simulations of spreading a rumor. We used six machine learning algorithms and four node embedding algorithms. The results, for three real world’s datasets, after combining machine learning and node embeddings, differs from method to method. Some of them are extraordinary and some of them aren’t that good.

**Keywords**: social networks, influential users, machine learning, classification, SI model, node embeddings

# Περιεχόμενα

[Ευχαριστίες 3](#_Toc107421570)

[Περίληψη 4](#_Toc107421571)

[Abstract 5](#_Toc107421572)

[Κατάλογος εικόνων 10](#_Toc107421574)

[Κατάλογος πινάκων 16](#_Toc107421575)

[Κεφάλαιο 1: Εισαγωγή 19](#_Toc107421576)

[Κεφάλαιο 2: Θεωρητικό Υπόβαθρο 21](#_Toc107421577)

[2.1. Θεωρία γράφων 21](#_Toc107421578)

[2.1.1 Ορισμός 21](#_Toc107421579)

[2.1.2 Κατηγορίες γράφων 22](#_Toc107421580)

[2.1.3 Πίνακας γειτνίασης (Adjacency matrix) 26](#_Toc107421581)

[2.1.4 Τάξη και μέγεθος γράφου 27](#_Toc107421582)

[2.1.5 Βαθμός κόμβου 27](#_Toc107421583)

[2.1.6 Μονοπάτι 28](#_Toc107421584)

[2.2 Embeddings 28](#_Toc107421585)

[2.2.1 Word2vec-Doc2vec 29](#_Toc107421586)

[2.2.2 Χρήση embeddings σε γράφους 30](#_Toc107421587)

[2.2.2.1 Κατηγορίες embeddings σε γράφους 30](#_Toc107421588)

[2.2.2.2 Διαφορετικές προσεγγίσεις δημιουργίας node embeddings 31](#_Toc107421589)

[2.2.2.3 Αλγόριθμοι δημιουργίας node embeddings 32](#_Toc107421590)

[2.2.2.4 Αλγόριθμος δημιουργίας graph embeddings 33](#_Toc107421591)

[2.3 Μηχανική Μάθηση 34](#_Toc107421592)

[2.3.1. Κατηγορίες προβλημάτων μηχανικής μάθησης 34](#_Toc107421593)

[2.3.1.1 Με βάση τη φύση του εκπαιδευτικού σήματος ή την ανατροφοδότηση που είναι διαθέσιμα σε ένα σύστημα εκμάθησης 35](#_Toc107421594)

[2.3.1.2 Με βάση το επιθυμητό αποτέλεσμα του συστήματος μηχανικής μάθησης 35](#_Toc107421595)

[2.4 Σύνολα δεδομένων που χρησιμοποιήθηκαν 38](#_Toc107421596)

[Κεφάλαιο 3: Μεθοδολογία 40](#_Toc107421597)

[3.1 Αξιολόγηση επιρροής κόμβων με την βοήθεια του μοντέλου επιδημίας SI 41](#_Toc107421598)

[3.1.1 Μοντέλο επιδημίας SI 41](#_Toc107421599)

[3.1.2 Αξιοποίηση μοντέλου SI 42](#_Toc107421600)

[3.1.3 Υπολογισμός επιρροής κάθε κόμβου 42](#_Toc107421601)

[3.1.4 Αλγόριθμος φυσικών ορίων του Jenks 43](#_Toc107421602)

[3.1.5 Υπολογισμός ετικετών 43](#_Toc107421603)

[3.2 Δημιουργία Node Embeddings 44](#_Toc107421604)

[3.2.1 Δημιουργία node embeddings με την χρήση της μεθόδου Node2vec 45](#_Toc107421605)

[3.2.2 Δημιουργία node embeddings με την χρήση της μεθόδου DeepWalk 47](#_Toc107421606)

[3.2.3 Δημιουργία node embeddings με την χρήση της μεθόδου Walklets 48](#_Toc107421607)

[3.2.4 Δημιουργία node embeddings με την χρήση μιας επέκτασης της μεθόδου Graph2vec για κόμβους (ext-Graph2vec) 50](#_Toc107421608)

[3.3 Σχηματισμός τελικού συνόλου δεδομένων εκπαίδευσης-αξιολόγησης 50](#_Toc107421609)

[3.4. Εκπαίδευση μοντέλων μηχανικής μάθησης 53](#_Toc107421610)

[3.4.1 Gaussian Naive Bayes 53](#_Toc107421611)

[3.4.2 Δέντρα απόφασης (Decision Tree) 54](#_Toc107421612)

[3.4.3 K-κοντινότεροι γείτονες (k-nearest neighbors) 54](#_Toc107421613)

[3.4.4 Λογιστική Παλινδρόμηση (Logistic Regression) 55](#_Toc107421614)

[3.4.5 Μηχανές Διανυσμάτων Υποστήριξης ΜΔΥ (Support Vector Machines, SVM) 55](#_Toc107421615)

[3.4.6 Νευρωνικό δίκτυο 56](#_Toc107421616)

[3.4.6.1 Επιλογή Παραμέτρων νευρωνικού δικτύου με την μέθοδο Hyperband 57](#_Toc107421617)

[3.5 Αξιολόγηση αποτελεσμάτων 58](#_Toc107421618)

[3.5.1 Cross Validation 58](#_Toc107421619)

[3.5.1.1 K-Fold cross validation 59](#_Toc107421620)

[3.5.2 Πίνακας σύγχυσης (Confusion Matrix) 59](#_Toc107421621)

[3.5.3 Μετρικές Ταξινόμησης 61](#_Toc107421622)

[3.5.3.1 Ευστοχία (accuracy) 61](#_Toc107421623)

[3.5.3.2 Ακρίβεια (precision) 62](#_Toc107421624)

[3.5.3.3 Ανάκληση (recall) 62](#_Toc107421625)

[3.5.3.4 F-score 62](#_Toc107421626)

[3.5.3.5 Μετρική για σύγκριση μεθόδων 62](#_Toc107421627)

[Κεφάλαιο 4: Αποτελέσματα 63](#_Toc107421628)

[4.1 Ταξινόμηση κόμβων συνδυάζοντας Node2vec και μηχανική μάθηση 63](#_Toc107421629)

[4.1.1 Κατασκευή δεδομένων εκπαίδευσης 64](#_Toc107421630)

[4.1.2 Απεικόνιση κόμβων και αλγόριθμοι ταξινόμησης 71](#_Toc107421631)

[4.1.3 Συμπεράσματα συνδυασμού Node2vec και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων 91](#_Toc107421632)

[4.2 Ταξινόμηση κόμβων συνδυάζοντας DeepWalk και μηχανική μάθηση 92](#_Toc107421633)

[4.2.1 Κατασκευή δεδομένων εκπαίδευσης 92](#_Toc107421634)

[4.2.2 DeepWalk και αλγόριθμοι ταξινόμησης 96](#_Toc107421635)

[4.2.3 Συμπεράσματα συνδυασμού DeepWalk και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων 117](#_Toc107421636)

[4.3 Ταξινόμηση κόμβων συνδυάζοντας Walklets και μηχανική μάθηση 117](#_Toc107421637)

[4.3.1 Κατασκευή δεδομένων εκπαίδευσης 117](#_Toc107421638)

[4.3.2 Walklets και αλγόριθμοι ταξινόμησης 121](#_Toc107421639)

[4.3.3 Συμπεράσματα συνδυασμού Walklets και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων 142](#_Toc107421640)

[4.4 Ταξινόμηση κόμβων συνδυάζοντας την επέκταση της μεθόδου Graph2vec για κόμβους (ext-Graph2vec) και τη μηχανική μάθηση 142](#_Toc107421641)

[4.4.1 Κατασκευή δεδομένων εκπαίδευσης 142](#_Toc107421642)

[4.3.2 Επέκταση μεθόδου Graph2vec και αλγόριθμοι ταξινόμησης 146](#_Toc107421643)

[4.4.3 Συμπεράσματα συνδυασμού της επέκτασης του Graph2vec για κόμβους και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων 167](#_Toc107421644)

[4.5 Σύγκριση αποτελεσμάτων 168](#_Toc107421645)

[4.5.1 Σύγκριση αλγορίθμων ταξινόμησης για το πρόβλεψη της επιρροής σε κοινωνικά δίκτυα 169](#_Toc107421646)

[4.5.2 Σύγκριση αλγορίθμων δημιουργίας node embeddings για την πρόβλεψη της επιρροής σε κοινωνικά δίκτυα 170](#_Toc107421647)

[Κεφάλαιο 5: Συμπεράσματα 172](#_Toc107421648)

[5.1. Ανασκόπηση εργασίας 172](#_Toc107421649)

[5.2 Μελλοντική έρευνα 173](#_Toc107421650)

[Βιβλιογραφία 174](#_Toc107421651)

# Κατάλογος εικόνων

[Εικόνα 1.1 Κοινωνικό Δίκτυο 19](#_Toc106182116)

[Εικόνα 2.1 Μη κατευθυνόμενος γράφος 21](#_Toc106182117)

[Εικόνα 2.2 Μηδενικός Γράφος 22](#_Toc106182118)

[Εικόνα 2.3 Κατευθυνόμενος Γράφος 23](#_Toc106182119)

[Εικόνα 2.4 Μη κατευθυνόμενος γράφος 23](#_Toc106182120)

[Εικόνα 2.5 Μεικτός γράφος 23](#_Toc106182121)

[Εικόνα 2.6 Πολυγράφος 24](#_Toc106182122)

[Εικόνα 2.7 Απλός γράφος 24](#_Toc106182123)

[Εικόνα 2.8 Σταθμισμένος γράφος 25](#_Toc106182124)

[Εικόνα 2.9 Πλήρης γράφος 25](#_Toc106182125)

[Εικόνα 2.10 Γράφος και υπογράφος 26](#_Toc106182126)

[Εικόνα 2.11 Συνεκτικός γράφος 26](#_Toc106182127)

[Εικόνα 2.12 Γράφος G μαζί με τον πίνακα γειτνίασής του 27](#_Toc106182128)

[Εικόνα 2.13 Παράδειγμα μονοπατιού σε έναν γράφο. 28](#_Toc106182129)

[Εικόνα 2.14 Word2vec 29](#_Toc106182130)

[Εικόνα 2.15 Παράδειγμα One Hot Encoding 30](#_Toc106182131)

[Εικόνα 2.16 Skip-gram 30](#_Toc106182132)

[Εικόνα 2.17 Παράδειγμα αναπαράστασης ενός γράφου με node embeddings δύο διαστάσεων 31](#_Toc106182133)

[Εικόνα 2.18 Βήματα DeepWalk 32](#_Toc106182134)

[Εικόνα 2.19 node2vec. 33](#_Toc106182135)

[Εικόνα 2.20 graph2vec 34](#_Toc106182136)

[Εικόνα 2.21 Παράδειγμα συσταδοποίησης 36](#_Toc106182137)

[Εικόνα 2.22 Δυαδική ταξινόμηση 37](#_Toc106182138)

[Εικόνα 2.23 Ταξινόμηση 3 κλάσεων 38](#_Toc106182139)

[Εικόνα 2.24 Multi-class vs Multi-label Classification 38](#_Toc106182140)

[Εικόνα 3.1 Διάγραμμα ροής μεθοδολογίας που ακολουθήθηκε 40](#_Toc106182141)

[Εικόνα 3.2 Το μοντέλο S-I 41](#_Toc106182142)

[Εικόνα 3.3 Παράδειγμα ενός από τα πεντακόσια spreads που δημιουργήθηκαν με την βοήθεια του μοντέλου SI 42](#_Toc106182143)

[Εικόνα 3.4 Παραδείγματα node embeddings 45](#_Toc106182144)

[Εικόνα 3.5 Δομή ταξινομητή 51](#_Toc106182145)

[Εικόνα 3.6 Παράδειγμα ταξινόμησης με την χρήση δέντρου απόφασης 54](#_Toc106182146)

[Εικόνα 3.7 k-nearest neighbors 55](#_Toc106182147)

[Εικόνα 3.8 Svm 56](#_Toc106182148)

[Εικόνα 3.9 Παράδειγμα νευρωνικού δικτύου 57](#_Toc106182149)

[Εικόνα 3.10 Overfitting. 58](#_Toc106182150)

[Εικόνα 3.11 Παράδειγμα K-Fold 59](#_Toc106182151)

[Εικόνα 3.12 Πίνακας σύγχυσης για προβλήματα δυαδικής ταξινόμησης 60](#_Toc106182152)

[Εικόνα 3.13 Πίνακας σύγχυσης για προβλήματα ταξινόμησης πολλών κλάσεων 61](#_Toc106182153)

[Εικόνα 4.1 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων CollegeMsg 64](#_Toc106182154)

[Εικόνα 4.2 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων Facebook. 64](#_Toc106182155)

[Εικόνα 4.3 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων Wiki-Votes. 65](#_Toc106182156)

[Εικόνα 4.4 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το CollegeMsg. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν 65](#_Toc106182157)

[Εικόνα 4.5 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το facebook. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν 65](#_Toc106182158)

[Εικόνα 4.6 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το Wiki-Votes. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν 66](#_Toc106182159)

[Εικόνα 4.7 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων CollegeMsg 66](#_Toc106182160)

[Εικόνα 4.8 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων Facebook 67](#_Toc106182161)

[Εικόνα 4.9 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων Wiki-Vote 67](#_Toc106182162)

[Εικόνα 4.10 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Node2vec για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής) 68](#_Toc106182163)

[Εικόνα 4.11 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Node2vec για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής) 69](#_Toc106182164)

[Εικόνα 4.12 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (3 κλάσεις επιρροής) 70](#_Toc106182165)

[Εικόνα 4.13 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (4 κλάσεις επιρροής) 71](#_Toc106182166)

[Εικόνα 4.14 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (3 κλάσεις επιρροής) 73](#_Toc106182167)

[Εικόνα 4.15 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (4 κλάσεις επιρροής) 74](#_Toc106182168)

[Εικόνα 4.16 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το CollegeMsg dataset (για 3 κλάσεις) 77](#_Toc106182169)

[Εικόνα 4.17 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το CollegeMsg dataset (για 4 κλάσεις) 80](#_Toc106182170)

[Εικόνα 4.18 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Facebook dataset (για 3 κλάσεις) 83](#_Toc106182171)

[Εικόνα 4.19 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Facebook dataset (για 4 κλάσεις) 86](#_Toc106182172)

[Εικόνα 4.20 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Wiki-Vote dataset (για 3 κλάσεις) 88](#_Toc106182173)

[Εικόνα 4.21 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Wiki-Vote dataset (για 4 κλάσεις) 91](#_Toc106182174)

[Εικόνα 4.22 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και DeepWalk για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής) 93](#_Toc106182175)

[Εικόνα 4.23 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και DeepWalk για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής) 94](#_Toc106182176)

[Εικόνα 4.24 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (3 κλάσεις επιρροής) 95](#_Toc106182177)

[Εικόνα 4.25 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (4 κλάσεις επιρροής) 96](#_Toc106182178)

[Εικόνα 4.26 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (3 κλάσεις επιρροής) 98](#_Toc106182179)

[Εικόνα 4.27 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (4 κλάσεις επιρροής) 99](#_Toc106182180)

[Εικόνα 4.28 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το CollegeMsg dataset (για 3 κλάσεις) 102](#_Toc106182181)

[Εικόνα 4.29 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το CollegeMsg dataset (για 4 κλάσεις) 105](#_Toc106182182)

[Εικόνα 4.30 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Facebook dataset (για 3 κλάσεις) 108](#_Toc106182183)

[Εικόνα 4.31 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Facebook dataset (για 4 κλάσεις) 110](#_Toc106182184)

[Εικόνα 4.32 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Wiki-Vote dataset (για 3 κλάσεις) 113](#_Toc106182185)

[Εικόνα 4.33 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Wiki-Vote dataset (για 4 κλάσεις) 116](#_Toc106182186)

[Εικόνα 4.34 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Walklets για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής) 118](#_Toc106182187)

[Εικόνα 4.35 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Walklets για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής) 119](#_Toc106182188)

[Εικόνα 4.36 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (3 κλάσεις επιρροής) 120](#_Toc106182189)

[Εικόνα 4.37 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (4 κλάσεις επιρροής) 121](#_Toc106182190)

[Εικόνα 4.38 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (3 κλάσεις επιρροής) 123](#_Toc106182191)

[Εικόνα 4.39 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (4 κλάσεις επιρροής) 124](#_Toc106182192)

[Εικόνα 4.40 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το CollegeMsg dataset (για 3 κλάσεις) 127](#_Toc106182193)

[Εικόνα 4.41 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το CollegeMsg dataset (για 4 κλάσεις) 130](#_Toc106182194)

[Εικόνα 4.42 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Facebook dataset (για 3 κλάσεις) 133](#_Toc106182195)

[Εικόνα 4.43 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Facebook dataset (για 4 κλάσεις) 135](#_Toc106182196)

[Εικόνα 4.44 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Wiki-Vote dataset (για 3 κλάσεις) 138](#_Toc106182197)

[Εικόνα 4.45 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Wiki-Vote dataset (για 4 κλάσεις) 141](#_Toc106182198)

[Εικόνα 4.46 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής) 143](#_Toc106182199)

[Εικόνα 4.47 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για τα CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής) 144](#_Toc106182200)

[Εικόνα 4.48 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (3 κλάσεις επιρροής) 145](#_Toc106182201)

[Εικόνα 4.49 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (4 κλάσεις επιρροής) 146](#_Toc106182202)

[Εικόνα 4.50 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (3 κλάσεις επιρροής) 148](#_Toc106182203)

[Εικόνα 4.51 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (4 κλάσεις επιρροής) 149](#_Toc106182204)

[Εικόνα 4.52 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το CollegeMsg dataset (για 3 κλάσεις) 152](#_Toc106182205)

[Εικόνα 4.53 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το CollegeMsg dataset (για 4 κλάσεις) 155](#_Toc106182206)

[Εικόνα 4.54 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Facebook dataset (για 3 κλάσεις) 158](#_Toc106182207)

[Εικόνα 4.55 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Facebook dataset (για 4 κλάσεις) 161](#_Toc106182208)

[Εικόνα 4.56 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Wiki-Vote dataset (για 3 κλάσεις) 164](#_Toc106182209)

[Εικόνα 4.57 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Wiki-Vote dataset (για 4 κλάσεις) 166](#_Toc106182210)

[Εικόνα 4.58 Αποτελεσματικότητα μεθόδου ταξινόμησης για κάθε αλγόριθμο δημιουργίας node embeddings για την περίπτωση των 3 κλάσεων 168](#_Toc106182211)

[Εικόνα 4.59 Αποτελεσματικότητα μεθόδου ταξινόμησης για κάθε αλγόριθμο δημιουργίας node embeddings για την περίπτωση των 7 κλάσεων 168](#_Toc106182212)

[Εικόνα 4.60 Διάγραμμα μέσης αποτελεσματικότητα ανά μέθοδο ταξινόμησης για 3 κλάσεις 169](#_Toc106182213)

[Εικόνα 4.61 Διάγραμμα μέσης αποτελεσματικότητα ανά μέθοδο ταξινόμησης για 3 κλάσεις 169](#_Toc106182214)

[Εικόνα 4.62 Μέση αποτελεσματικότητα ταξινόμησης για κάθε μέθοδο δημιουργίας node embeddings που χρησιμοποιήθηκε για 3 κλάσεις επιρροής 170](#_Toc106182215)

[Εικόνα 4.63 Μέση αποτελεσματικότητα ταξινόμησης για κάθε μέθοδο δημιουργίας node embeddings που χρησιμοποιήθηκε για 4 κλάσεις επιρροής 171](#_Toc106182216)

# Κατάλογος πινάκων

[Πίνακας 2.1 Σύνολα δεδομένων 39](#_Toc106182217)

[Πίνακας 3.1 Κατανομή παραδειγμάτων ανά κλάση για την περίπτωση των 3 κλάσεων 44](#_Toc106182218)

[Πίνακας 3.2 Κατανομή παραδειγμάτων ανά κλάση για την περίπτωση των 4 κλάσεων 44](#_Toc106182219)

[Πίνακας 3.3 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου Node2vec 47](#_Toc106182220)

[Πίνακας 3.4 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου DeepWalk 48](#_Toc106182221)

[Πίνακας 3.5 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου Walklets 50](#_Toc106182222)

[Πίνακας 3.6 Παράμετροι node embeddings για κάθε dataset για κάθε μέθοδο δημιουργίας node embeddings (3 κλάσεις) 52](#_Toc106182223)

[Πίνακας 3.7 Παράμετροι node embeddings για κάθε dataset dataset για κάθε μέθοδο δημιουργίας node embeddings (4 κλάσεις) 52](#_Toc106182224)

[Πίνακας 4.1 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset CollegeMsg (3 κλάσεις) 75](#_Toc106182225)

[Πίνακας 4.2 Αποτελέσματα για συνδυασμό node2vec με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις) 78](#_Toc106182226)

[Πίνακας 4.3 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Facebook (3 κλάσεις) 81](#_Toc106182227)

[Πίνακας 4.4 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Facebook (4 κλάσεις) 84](#_Toc106182228)

[Πίνακας 4.5 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Wiki-Vote (3 κλάσεις) 87](#_Toc106182229)

[Πίνακας 4.6 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Wiki-Vote (4 κλάσεις) 89](#_Toc106182230)

[Πίνακας 4.7 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις) 100](#_Toc106182231)

[Πίνακας 4.8 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις) 103](#_Toc106182232)

[Πίνακας 4.9 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Facebook (3 κλάσεις) 106](#_Toc106182233)

[Πίνακας 4.10 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Facebook (4 κλάσεις) 108](#_Toc106182234)

[Πίνακας 4.11 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις) 112](#_Toc106182235)

[Πίνακας 4.12 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις) 114](#_Toc106182236)

[Πίνακας 4.13 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις) 125](#_Toc106182237)

[Πίνακας 4.14 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις) 128](#_Toc106182238)

[Πίνακας 4.15 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Facebook (3 κλάσεις) 131](#_Toc106182239)

[Πίνακας 4.16 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Facebook (4 κλάσεις) 133](#_Toc106182240)

[Πίνακας 4.17 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις) 137](#_Toc106182241)

[Πίνακας 4.18 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις) 139](#_Toc106182242)

[Πίνακας 4.19 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις) 150](#_Toc106182243)

[Πίνακας 4.20 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις) 153](#_Toc106182244)

[Πίνακας 4.21 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Facebook (3 κλάσεις) 156](#_Toc106182245)

[Πίνακας 4.22 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Facebook (4 κλάσεις) 159](#_Toc106182246)

[Πίνακας 4.23 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις) 162](#_Toc106182247)

[Πίνακας 4.24 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις) 164](#_Toc106182248)

# Κεφάλαιο 1: Εισαγωγή

Τα κοινωνικά δίκτυα ορίζονται ως σύνολα αλληλεπιδράσεων και διαπροσωπικών σχέσεων. Στις μέρες μας, ο όρος αυτός χρησιμοποιείται, κατά βάση, για να περιγράψει ιστοσελίδες και ψηφιακές εφαρμογές, οι οποίες δίνουν την δυνατότητα στους χρήστες να σχηματίσουν κοινωνικές σχέσεις μεταξύ τους με διάφορα κριτήρια όπως τα κοινά ενδιαφέροντα, η γνωριμία στον πραγματικό κόσμο κ.α. Η αναπαράσταση των δικτύων γίνεται με την βοήθεια των κόμβων και των ακμών, όπου κόμβοι είναι οι χρήστες του δικτύου και ακμές οι σχέσεις μεταξύ τους. Μέσω αυτών των σχέσεων πραγματοποιείται διακίνηση ιδεών, συναισθημάτων, συμπεριφορών, οπότε μπορούμε να πούμε ότι μεταξύ των χρηστών ασκείται επιρροή. Η κοινωνική επιρροή έχει εφαρμογές σε πολλούς τομείς όπως στην διάδοση ειδήσεων, στην προώθηση προϊόντων και υπηρεσιών καθώς και στην πολιτική γεγονός που οδηγεί στην αύξηση του ερευνητικού-επιστημονικού ενδιαφέροντος γύρω από αυτή.

Εικόνα που περιέχει ουρανός, μέταλλο, λάμπα

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 1.1 Κοινωνικό Δίκτυο

Η μελέτη της κοινωνικής επιρροής αποτελεί ένα από τα σημαντικότερα ερευνητικά πεδία στο ευρύτερο πεδίο των κοινωνικών δικτύων. Ασχολείται, κατά βάση, με την μελέτη της διαδικασίας διάδοσης ιδεών σε ένα δίκτυο καθώς και με την δημιουργία μοντέλων εύρεσης και ανάλυσης της επιρροής των. Η επιρροή ενός χρήστη μελετάται με πολλούς διαφορετικούς τρόπους όπως το πλήθος των χρηστών με τους οποίους αλληλοεπιδρά, το πλήθος των χρηστών που υιοθετούν μια ιδέα που ξεκίνησε από αυτόν, από την πιθανότητα συμμετοχής του στην διάδοση μιας ιδέας ανεξαρτήτως από πού ξεκίνησε αυτή. Σημαντικό παράδειγμα εφαρμογής της μελέτης αυτής αποτελεί το marketing. Μια εταιρία, για να μειώσει το κόστος διαφήμισης ενός προϊόντος χρησιμοποιεί διάφορες τεχνικές εντοπισμού των ατόμων εκείνων που θα επηρεάσουν θετικά μεγαλύτερο κομμάτι του πληθυσμού. Παράλληλα, τέτοια μοντέλα υιοθετούνται και σε άλλα δίκτυα που εξετάζονται για την μετάδοση επιδημιών. Με τον εντοπισμό των ατόμων που μεταδίδουν σε μεγαλύτερο κομμάτι του πληθυσμού την επιδημία, καθώς και με τον καθορισμό των ξεχωριστών χαρακτηριστικών αυτών που τους καθιστά ευάλωτους σε αυτή γίνεται πιο εύκολος ο έλεγχος της εξάπλωσης της.

Ο στόχος της παρούσας διπλωματικής εργασίας είναι η πρόβλεψη της επιρροής σε ένα κοινωνικό δίκτυο. Η μεθοδολογία που ακολουθήθηκε, αντιμετωπίζει το συγκεκριμένο πρόβλημα από την σκοπιά της ταξινόμησης των χρηστών με βάση την επιρροή τους [1] [2]. Αρχικά πραγματοποιήθηκε αξιολόγηση της επιρροής των κόμβων. Σε αντίστοιχες μελέτες η αξιολόγηση προκύπτει με βάση τις μετρικές κεντρικότητας, οι οποίες είναι μεγέθη που δείχνουν τοπολογικά στοιχεία του δικτύου όπως το πόσο διασυνδεδεμένος είναι κάθε κόμβος με τους υπόλοιπους. Στην συγκεκριμένη διπλωματική εργασία για την αξιολόγηση της επιρροής κάθε κόμβου χρησιμοποιήθηκε το επιδημιολογικό μοντέλο SI (Susceptible-Infected) [3]. Στην συνέχεια για να γίνει η ταξινόμηση των κόμβων σε ομάδες με κριτήριο την επιρροή τους, έπρεπε για κάθε κόμβο να δημιουργηθεί μια αναπαράσταση, η οποία μπορεί να χρησιμοποιηθεί από τα μοντέλα ταξινόμησης. Για να σχηματιστούν οι αναπαραστάσεις αυτές, επιλέχθηκε η χρησιμοποίηση αλγορίθμων δημιουργίας node embeddings. Στην παρούσα εργασία για να σχηματιστούν τα node embeddings χρησιμοποιήθηκαν τέσσερις τέτοιοι αλγόριθμοι έτσι ώστε να προκύψει μια σύγκριση μεταξύ των αποτελεσμάτων τους πάνω στην συγκεκριμένη ταξινόμηση. Τέλος, γίνεται η πρόβλεψη της επιρροής των κόμβων, τριών πραγματικών κοινωνικών δικτύων, με την βοήθεια έξι διαφορετικών μεθόδων ταξινόμησης.

Στην παρούσα διπλωματική εργασία ακολουθείται η εξής δομή: Στο κεφάλαιο 2, παρουσιάζονται λεπτομέρειες σχετικά με τη θεωρία των γραφημάτων, την χρήση embeddings για την αναπαράσταση μεμονωμένων κόμβων και γράφων, καθώς και άλλες βασικές έννοιες, που σχετίζονται με το θέμα της πρόβλεψης της επιρροής στα κοινωνικά δίκτυα. Οι προτεινόμενοι μέθοδοι και τα διάφορα χαρακτηριστικά τους περιγράφονται με λεπτομέρειες στο κεφάλαιο 3. Στο κεφάλαιο 4, παρουσιάζονται τα πειράματα που διεξάγονται σε τρία πραγματικά σύνολα δεδομένων και συζητούνται τα αποτελέσματά τους. Τέλος, το κεφάλαιο 5 ολοκληρώνει το έργο της εργασίας, με μια σύντομη επισκόπηση των προτεινόμενων μεθόδων και επισημαίνει ορισμένες κατευθύνσεις για μελλοντικές εργασίες.

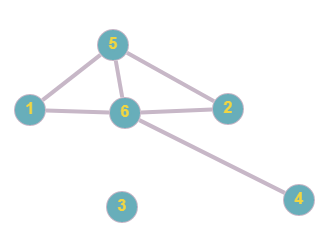
# Κεφάλαιο 2: Θεωρητικό Υπόβαθρο

Στο κεφάλαιο αυτό παρουσιάζονται όλες οι θεμελιώδεις έννοιες που σχετίζονται με το πρόβλημα της πρόβλεψης επιρροής, με την χρήση embeddings, σε κοινωνικά δίκτυα και γράφους. Αρχικά γίνεται μία παρουσίαση της θεωρίας των γράφων. Στην συνέχεια αναλύεται η θεωρία γύρω από τα embeddings δίνοντας μεγαλύτερη έμφαση στα node embeddings και στα graph embeddings που σχετίζονται με τους γράφους. Τέλος μετά από την περιγραφή του πεδίου της μηχανικής μάθησης παρουσιάζονται τα σύνολα δεδομένων που χρησιμοποιήθηκαν κατά την μεθοδολογία.

## 2.1. Θεωρία γράφων

### 2.1.1 Ορισμός

Ο γράφος ορίζεται ως η οπτική αναπαράσταση ενός συνόλου σημείων, μερικά εκ των οποίων ενώνονται μεταξύ τους με γραμμές. Τα σημεία αυτά ονομάζονται **κόμβοι** (ή κορυφές) ενώ οι γραμμές ονομάζονται **ακμές**. Ένας γράφος G μπορεί να αναπαρασταθεί ως ζεύγος G = (V,E), όπου V ή V (G) είναι το σύνολο των κορυφών ή των κόμβων και E ή E(G) είναι το σύνολο των ακμών ή των συνδέσεων μεταξύ των κόμβων. Όταν δύο κόμβοι ενώνονται μέσω μίας ακμής, τότε λέμε ότι οι κόμβοι αυτοί είναι γειτονικοί. Σε πολλές περιπτώσεις οι ακμές που ενώνουν τους κόμβους μπορούν να έχουν συγκεκριμένη κατεύθυνση από τον έναν κόμβο προς τον άλλον. Αυτή η κατεύθυνση δηλώνεται με την χρήση βέλους αντί απλής γραμμής. Γενικότερα, η χρήση γράφων βρίσκει πολλές εφαρμογές στον πραγματικό κόσμο όπως, στην αναπαράσταση κοινωνικών δικτύων, την αναπαράσταση οδικών δικτύων, στον παγκόσμιο ιστό, σε δρομολόγηση και ανάθεση πόρων κ.α.



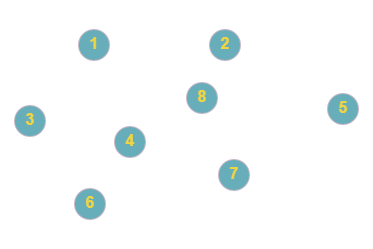
Εικόνα 2.1 Μη κατευθυνόμενος γράφος

### 2.1.2 Κατηγορίες γράφων

Υπάρχουν πολλές διαφορετικές κατηγορίες γράφων με βάση τον αριθμό των κόμβων και των ακμών, την συνδεσιμότητα και την ολική δομή τους. Μερικές από αυτές τις κατηγορίες αναφέρονται παρακάτω

* Μηδενικός γράφος

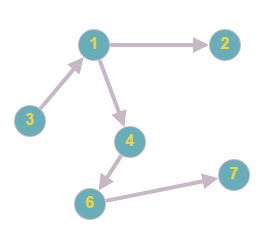
Ένας μηδενικός γράφος αποτελείται από n απομονωμένους κόμβους μεταξύ των οποίων δεν υπάρχουν ακμές.



Εικόνα 2.2 Μηδενικός Γράφος

* Κατευθυνόμενος γράφος

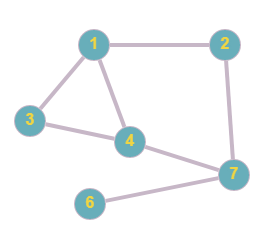
Ένας γράφος ονομάζεται κατευθυνόμενος όταν κάθε ακμή του είναι προσανατολισμένη προς μία κατεύθυνση.



Εικόνα 2.3 Κατευθυνόμενος Γράφος

* Μη κατευθυνόμενος γράφος

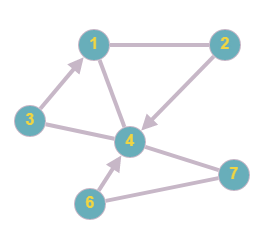
Ένας γράφος ονομάζεται μη κατευθυνόμενος όταν οι ακμές του δεν είναι προσανατολισμένες προς μία κατεύθυνση.



Εικόνα 2.4 Μη κατευθυνόμενος γράφος

* Μεικτός γράφος

Μεικτός γράφος είναι ένας γράφος στον οποίο μερικές ακμές είναι κατευθυνόμενες και άλλες είναι μη κατευθυνόμενες.



Εικόνα 2.5 Μεικτός γράφος

* Πολυγράφος

Πολυγράφος είναι ένας γράφος στον οποίο επιτρέπονται πολλαπλές ακμές μεταξύ του ίδιου ζεύγους κόμβων καθώς και βρόχοι (ακμές που ξεκινούν και τελειώνουν στον ίδιο κόμβο.

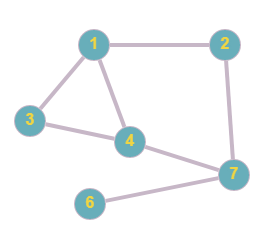
Εικόνα που περιέχει αξεσουάρ

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 2.6 Πολυγράφος

* Απλός γράφος

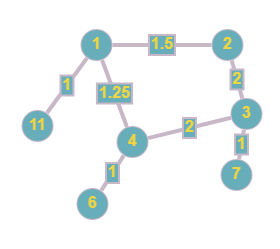
Απλός γράφος είναι ένας μη κατευθυνόμενος γράφος που δεν έχει βρόχους και έχει όχι περισσότερες από μία ακμή ανάμεσα σε δύο διαφορετικές κορυφές



Εικόνα 2.7 Απλός γράφος

* Σταθμισμένος γράφος

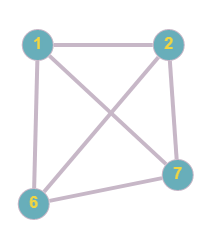
Σε κάθε ακμή του σταθμισμένου γράφου ανατίθεται ένας αριθμός (βάρος).



Εικόνα 2.8 Σταθμισμένος γράφος

* Πλήρης γράφος

Ένας πλήρης γράφος έχει το χαρακτηριστικό ότι κάθε ζευγάρι κορυφών έχει μια ακμή που να τους συνδέει



Εικόνα 2.9 Πλήρης γράφος

* Υπογράφος

Ένας γράφος G1 είναι υπογράφος του γράφου G αν αποτελείται από ένα υποσύνολο των ακμών και των κορυφών του G, έτσι ώστε οι τελικοί κόμβοι των ακμών στο G1 να βρίσκονται επίσης στο G.

|  |  |
| --- | --- |
|  |  |

Εικόνα 2.10 Γράφος και υπογράφος

* Συνεκτικός γράφος

Ένας μη κατευθυνόμενος γράφος θεωρείται συνεκτικός όταν για κάθε ζευγάρι κόμβων υπάρχει διαδρομή που τους συνδέει.

Εικόνα που περιέχει άτομο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 2.11 Συνεκτικός γράφος

### 2.1.3 Πίνακας γειτνίασης (Adjacency matrix)

Αν το ζεύγος (u,v) αποτελεί ακμή μεταξύ των κόμβων u και v τότε λέμε ότι οι κόμβοι u και v είναι γειτονικοί ή ότι γειτνιάζουν. Ένας τρόπος αναπαράστασης ενός γράφου είναι ο πίνακας γειτνίασης, όπου ένας γράφος G(V,E) αναπαρίσταται ως ένας δισδιάστατος πίνακας Α. Κάθε γραμμή και κάθε στήλη απευθύνεται σε έναν κόμβο. Στο παρακάτω παράδειγμα της εικόνας 13 υπάρχει ακμή μεταξύ του κόμβου 1 και του κόμβου 2 όποτε στις θέσεις (1,2) και (2,1) του πίνακα τοποθετείται το 1 (στην περίπτωση των σταθμισμένων γράφων αντί για το 1 τοποθετείται το βάρος της αντίστοιχης ακμής).

|  |  |
| --- | --- |
| Εικόνα που περιέχει μπάλα μπιλιάρδου, σκούρος  Περιγραφή που δημιουργήθηκε αυτόματα | Εικόνα που περιέχει κείμενο, μαγειρικά σκεύη  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 2.12 Γράφος G μαζί με τον πίνακα γειτνίασής του

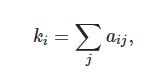
### 2.1.4 Τάξη και μέγεθος γράφου

Τάξη: Ο αριθμός των κορυφών ενός γράφου G(V, E) ονομάζεται τάξη (order) του γράφου G και συμβολίζεται με |V|

Μέγεθος: Ο αριθμός των ακμών ενός γράφου G(V, E), ονομάζεται μέγεθος (size) του γράφου G και συμβολίζεται με |E|.

### 2.1.5 Βαθμός κόμβου

Στη θεωρία των γράφων, ο βαθμός ενός κόμβου είναι ο αριθμός των ακμών που εφάπτονται με τον κόμβο, δηλαδή το σύνολο των γειτόνων του. Σε έναν πολυγράφο, ένας βρόχος συνεισφέρει 2 στον βαθμό ενός κόμβου για τα δύο άκρα της ακμής. Σε έναν μη κατευθυνόμενο γράφο ο βαθμός ενός κόμβου i δίνεται από



όπου j το πλήθος των κόμβων του γράφου . Αντίστοιχα σε έναν κατευθυνόμενο γράφο κάθε κόμβος έχει 2 βαθμούς, έναν για τις ακμές που κατευθύνονται προς αυτόν και έναν για τις ακμές που κατευθύνονται από αυτόν. Ο συνολικός βαθμός δίνεται από το άθροισμα των δύο παραπάνω βαθμών.

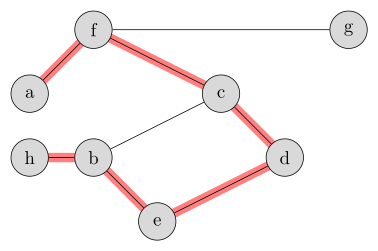
|  |  |
| --- | --- |
| Εικόνα που περιέχει κείμενο, ρολόι  Περιγραφή που δημιουργήθηκε αυτόματα | Εικόνα που περιέχει κείμενο, ρολόι  Περιγραφή που δημιουργήθηκε αυτόματα |



Αποδεικνύεται εύκολα ότι το άθροισμα των βαθμών όλων των κορυφών ενός γράφου, ισούται αριθμητικά με το διπλάσιο του αριθμού των ακμών του

### 2.1.6 Μονοπάτι

Μονοπάτι ή διαδρομή (path) ενός γράφου μήκους n, είναι μια ακολουθία κόμβων v0, v1, …, vn, όπου για κάθε i , 0≤ i < n, (vi , vi+1) είναι ακμή του γράφου. Μήκος ενός μονοπατιού είναι ο αριθμός ακμών που περιέχει.



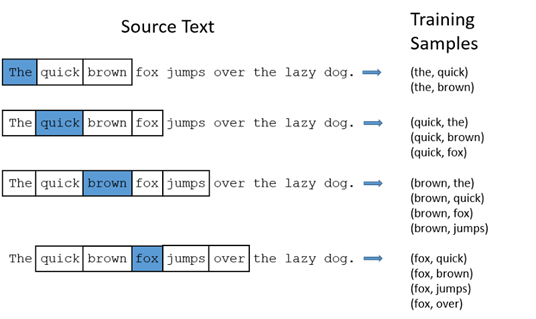
Εικόνα 2.13 Παράδειγμα μονοπατιού σε έναν γράφο.

## 2.2 Embeddings

Τα embeddings είναι μια τεχνική αναπαράστασης σύνθετων αντικειμένων όπως λέξεων, κειμένων, γράφων κ.α. , η οποία χρησιμοποιείται κυρίως στην Μηχανική Μάθηση. Στόχος αυτής της τεχνικής, είναι η μετατροπή των σύνθετων αντικειμένων σε διανύσματα (vectors) αριθμών, συγκεκριμένων διαστάσεων, τα οποία θα αντιπροσωπεύουν την πραγματική σχέση των αντικειμένων αυτών, δηλαδή θα διατηρούν τις σημαντικότερες τους πληροφορίες. Για έναν υπολογιστή είναι ευκολότερο να δουλέψει με embeddings, γιατί έτσι είναι πιο εύκολος ο υπολογισμός της ομοιότητας (πόσο κοντά είναι στον χώρο) 2 αντικειμένων. Η πιο συνηθισμένη μετρική ομοιότητας που χρησιμοποιείται είναι η cosine similarity [4].

### 2.2.1 Word2vec-Doc2vec

Το Word2vec [5] είναι μια μέθοδος μετατροπής λέξεων σε διανύσματα αριθμών (embeddings). Παρόμοιες λέξεις έχουν παρόμοια embeddings. Σε αυτή την μέθοδο χρησιμοποιείται το skip-gram, ένα νευρωνικό δίκτυο με ένα κρυφό επίπεδο, το οποίο εκπαιδεύεται να προβλέπει γειτονικές λέξεις σε μία πρόταση. Το δίκτυο δέχεται σαν είσοδο την λέξη (σε μορφή one hot encoding) και προβλέπει την γειτονική λέξη της πρότασης με μεγάλη πιθανότητα [6]. To One Hot Encoding είναι ένα διάνυσμα αναπαράστασης με μήκος όσο το μήκος του λεξικού που χρησιμοποιείται. Σε αυτό το διάνυσμα όλες οι τιμές είναι 0, εκτός της θέσης που βρίσκεται η λέξη της εισόδου, στην οποία μπαίνει το 1. Σε εντελώς αντίστοιχη λογική λειτουργεί και ο αλγόριθμος doc2vec [7] για ολόκληρες προτάσεις και έγγραφα. Τα Word2vec και doc2vec καθώς και το skip-gram αποτελούν την βάση για τα embeddings σε γράφους [6].

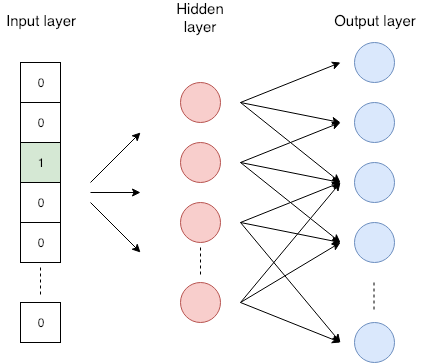


Εικόνα 2.14 Word2vec

Εικόνα που περιέχει πίνακας

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 2.15 Παράδειγμα One Hot Encoding



Εικόνα 2.16 Skip-gram

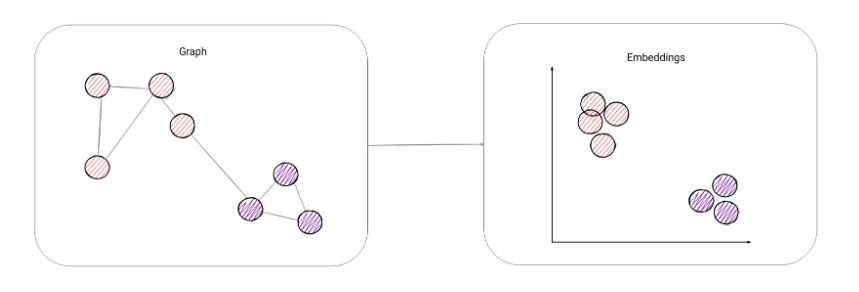
### 2.2.2 Χρήση embeddings σε γράφους

Με την τεχνική των embeddings επιτυγχάνεται η αναπαράσταση των κόμβων ή ολόκληρων γράφων σε μορφή που μπορεί να χρησιμοποιηθεί από τον υπολογιστή [12].

#### 2.2.2.1 Κατηγορίες embeddings σε γράφους

Υπάρχουν 2 διαφορετικοί τρόποι απεικόνισης που χρησιμοποιούνται στους γράφους:

α) **Η απεικόνιση κόμβου (node embedding)**: Είναι η τεχνική με την οποία για κάθε κόμβο ενός γράφου δημιουργείται ένα embedding, με συγκεκριμένο αριθμό διαστάσεων. Η σχέση μεταξύ των κόμβων στον πραγματικό γράφο φαίνεται στην ομοιότητα (όποια συνάρτηση ομοιότητας κι αν επιλεγεί) των διανυσμάτων. Τα node embeddings χρησιμοποιούνται σε προβλήματα πρόβλεψης σε επίπεδο χρήστη, όπως ταξινόμηση κόμβων, ομαδοποίηση κόμβων, πρόβλεψη ακμών κ.α.



Εικόνα 2.17 Παράδειγμα αναπαράστασης ενός γράφου με node embeddings δύο διαστάσεων

β) **Η απεικόνιση ολόκληρου γράφου (graph embedding)**: Είναι η τεχνική με την οποία δημιουργείται ένα embedding για έναν ολόκληρο γράφο. Τα graph embeddings χρησιμοποιούνται σε προβλήματα πρόβλεψης σε επίπεδο γράφου, όπως η σύγκριση χημικών μορίων.

#### 2.2.2.2 Διαφορετικές προσεγγίσεις δημιουργίας node embeddings

Υπάρχουν 3 διαφορετικές προσεγγίσεις δημιουργίας node embeddings:

α)**Factorization based methods**: Εδώ ανήκουν αλγόριθμοι που αναπαριστούν τις συνδέσεις μεταξύ κόμβων σε μορφή πίνακα και παραγοντοποιούν τον πίνακα αυτό για να αποκτήσουν τα embeddings. Σε μια τέτοια μέθοδο, που ονομάζεται *Local Linear Embedding*, γίνεται η υπόθεση ότι κάθε κόμβος είναι ένας γραμμικός συνδυασμός των γειτόνων του. Στη συνέχεια, ο αλγόριθμος προσπαθεί να αναπαραστήσει το embedding κάθε κόμβου σαν γραμμικό συνδυασμό των embeddings των γειτόνων του. [4]

β) **Random walk based methods**: Σε αυτή την κατηγορία ανήκουν αλγόριθμοι που χρησιμοποιούν την προσέγγιση των τυχαίων περιπάτων για να παράξουν γειτονιές από κόμβους σε ένα δίκτυο. Για κάθε κόμβο, παράγεται μια τέτοια γειτονιά επιλέγοντας με κάποιο τρόπο τον επόμενο κόμβο του περιπάτου. Αυτή η επιλογή εξαρτάται από τον αλγόριθμο. Το μέγιστο μήκος του περιπάτου (walk\_length) καθορίζεται στην αρχή σαν παράμετρος και για κάθε κόμβο παράγονται Ν τυχαίοι περίπατοι (random walks). Με αυτόν τον τρόπο έχει δημιουργηθεί μια γειτονιά για κάθε κόμβο, όποτε επόμενος στόχος θα έπρεπε να είναι, το να κάνουμε κάθε κόμβο όσο το δυνατόν πιο όμοιο με τους υπόλοιπους κόμβους της γειτονιάς του. Η ομοιότητα αυτή όμως προκύπτει από την πιθανότητα δύο κόμβοι να συνυπάρχουν σε έναν τυχαίο περίπατο. [4]

γ) **Deep learning methods**: Σε αυτή την κατηγορία ανήκουν αλγόριθμοι, που χρησιμοποιούν μεθόδους βασισμένες στα νευρωνικά δίκτυα (Deep learning). Με τα νευρωνικά δίκτυα είναι πιο εύκολη η μοντελοποίηση μη γραμμικών δομών, οπότε οι deep autoencoders χρησιμοποιούνται για μείωση των διαστάσεων των embeddings.

#### 2.2.2.3 Αλγόριθμοι δημιουργίας node embeddings

α) **DeepWalk** : Το DeepWalk [8] είναι ένας αλγόριθμος που χρησιμοποιεί random walks για την παραγωγή embeddings. Ο τυχαίος περίπατος ξεκινά από έναν επιλεγμένο κόμβο και συνεχίζει προς ένα τυχαίο γείτονα για έναν ορισμένο αριθμό βημάτων. Η μέθοδος αποτελείται από 3 βήματα [6]

* Sampling: Σε αυτό το βήμα γίνονται μερικοί τυχαίοι περίπατοι για κάθε κόμβο
* Εκπαίδευση skip-gram: Οι τυχαίοι περίπατοι είναι κάτι σαν τις προτάσεις στο Word2vec. Σε αυτή την περίπτωση το skip-gram δέχεται σαν είσοδο ένα κόμβο του τυχαίου περίπατου (με την μορφή διανύσματος One-Hot Encoding) και μεγιστοποιεί την πιθανότητα να προβλέπει γειτονικούς κόμβους.
* Υπολογισμός embeddings: Σε αυτό το βήμα υπολογίζονται τα embeddings

Εικόνα που περιέχει κείμενο, όπλο, σιδερογροθιά, ψαλίδι

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 2.18 Βήματα DeepWalk

β) **Node2vec**: Το Node2vec [9] είναι μια παραλλαγή του DeepWalk με την διαφορά μεταξύ τους να εμφανίζεται στο κομμάτι των random walks. Στο DeepWalk οι τυχαίοι περίπατοι είναι εντελώς τυχαίοι. O Node2vec, αντίθετα, έχει 2 επιπλέον παραμέτρους, τα P και Q. Η παράμετρος Q, καθορίζει πόσο πιθανό είναι το random walk να ανακαλύψει το κομμάτι του γράφου στο οποίο δεν έχει μεταβεί, ενώ η παράμετρος P καθορίζει πόσο πιθανό είναι να επιστρέψει σε προηγούμενο κόμβο [6]. Τα υπόλοιπα βήματα των αλγορίθμων είναι ίδια.

Εικόνα που περιέχει κείμενο, ρολόι

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 2.19 Η εικόνα δείχνει τις πιθανότητες ενός βήματος ενός random walk. Μόλις έχει γίνει η μετάβαση από τον κόμβο t στον v. Η πιθανότητα να ξαναπάμε στον t είναι 1/p, η πιθανότητα να πάμε σε κόμβο που δεν είναι σε επαφή με τον t είναι 1/q ενώ η πιθανότητα να πάμε σε κόμβο που είναι σε επαφή με τον t είναι 1.

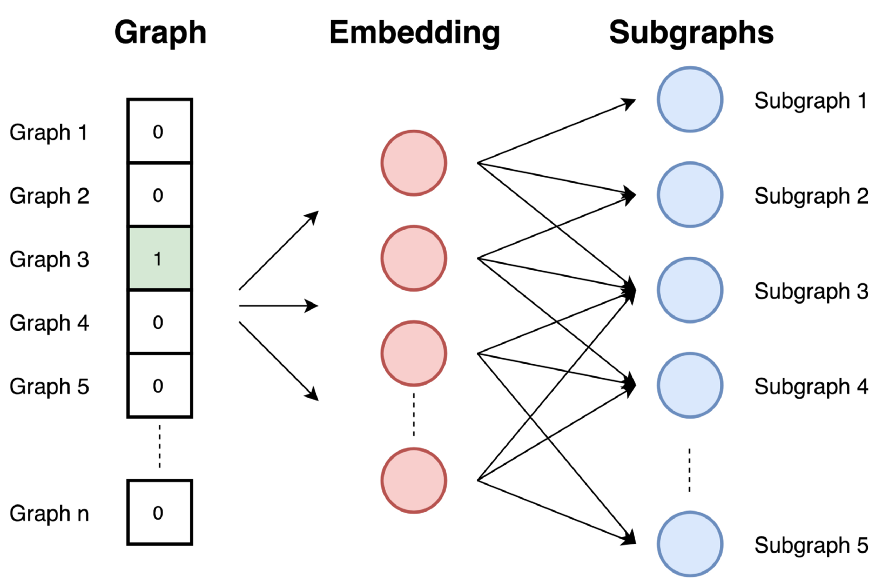
γ) **Walklets**: Το Walklets [10] είναι ένας αλγόριθμος που χρησιμοποιεί random walks για την παραγωγή embeddings και ουσιαστικά αποτελεί επέκταση του DeepWalk. Η διαφορά μεταξύ τους είναι ότι στο πρώτο βήμα, δηλαδή στο Sampling, γίνεται η επιλογή μερικοί κόμβοι να προσπεραστούν στους τυχαίους περιπάτους. Με αυτή την μέθοδο δημιουργείται ένα σώμα με ζεύγη κόμβων που είναι προσβάσιμα μέσω μονοπατιών σταθερού μήκους. Αυτό το σώμα χρησιμοποιείται για την εκπαίδευση των embeddings.

#### 2.2.2.4 Αλγόριθμος δημιουργίας graph embeddings

**Graph2vec**: Το Graph2vec [11] είναι ένας αλγόριθμος που στόχο έχει την δημιουργία embeddings για ολόκληρους γράφους και όχι για κόμβους. Βασίζεται στην ιδέα του doc2vec που χρησιμοποιεί δίκτυο skip-gram (τo doc2vec παίρνει το ID ενός εγγράφου σαν είσοδο και εκπαιδεύεται να μεγιστοποιεί την πιθανότητα πρόβλεψης τυχαίων λέξεων από το έγγραφο) [6]. Η μέθοδος Graph2vec αποτελείται από 3 βήματα:

* Εξαγωγή υπογράφων (sub-graph) από τον γράφο: Το sub-graph είναι ένα σετ κόμβων που εμφανίζονται γύρω από έναν επιλεγμένο κόμβο.
* Εκπαίδευση του skip-gram: Οι γράφοι είναι παρόμοιοι με τα έγγραφα. Όπως τα έγγραφα είναι σύνολα λέξεων, οι γράφοι είναι σύνολα υπογράφων. Σε αυτό το βήμα, το skip-gram εκπαιδεύεται να μεγιστοποιεί την πιθανότητα πρόβλεψης της ύπαρξης του υπογράφου στον γράφο που δίνεται στην είσοδο. Ο γράφος της εισόδου δίνεται σαν διάνυσμα One Hot.
* Υπολογισμός Embeddings: Σε αυτό το βήμα υπολογίζεται το embedding του γράφου της εισόδου.

Εφόσον γίνεται πρόβλεψη υπογράφων καταλαβαίνουμε ότι οι γράφοι με παρόμοιους υπογράφους και παρόμοια δομή έχουν παρόμοια embeddings.



Εικόνα 2.20 graph2vec

## 2.3 Μηχανική Μάθηση

Η μηχανική μάθηση είναι υποπεδίο της επιστήμης των υπολογιστών που αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη. Η μηχανική μάθηση διερευνά τη μελέτη και την κατασκευή αλγορίθμων που μπορούν να μαθαίνουν από τα δεδομένα και να κάνουν προβλέψεις σχετικά με αυτά [12]. Τέτοιοι αλγόριθμοι λειτουργούν κατασκευάζοντας μοντέλα από πειραματικά δεδομένα, προκειμένου να κάνουν προβλέψεις βασιζόμενες στα δεδομένα ή να εξάγουν αποφάσεις που εκφράζονται ως το αποτέλεσμα [12]. Με άλλα λόγια, οι αλγόριθμοι μηχανικής μάθησης χρησιμοποιούν στατιστικά στοιχεία για την εύρεση σχεδίων ή μοτίβων μεγάλα σύνολα δεδομένων. Τα δεδομένα αυτά μπορούν να είναι σε πολλές μορφές όπως λέξεις, εικόνες, αριθμοί κ.α. Οτιδήποτε μπορεί να αποθηκευτεί ψηφιακά, τότε μπορεί να χρησιμοποιηθεί σε έναν αλγόριθμο μηχανικής μάθησης.

### 2.3.1. Κατηγορίες προβλημάτων μηχανικής μάθησης

#### 2.3.1.1 Με βάση τη φύση του εκπαιδευτικού σήματος ή την ανατροφοδότηση που είναι διαθέσιμα σε ένα σύστημα εκμάθησης

* **Επιτηρούμενη μάθηση** (αλλιώς επιβλεπόμενη μάθηση ή μάθηση με επίβλεψη) (supervised learning)

Το υπολογιστικό πρόγραμμα δέχεται τις παραδειγματικές εισόδους καθώς και τα επιθυμητά αποτελέσματα από έναν «δάσκαλο», και ο στόχος είναι να μάθει έναν γενικό κανόνα προκειμένου να αντιστοιχίσει τις εισόδους με τα αποτελέσματα [12]. Με άλλα λόγια, τα σύνολα δεδομένων που χρησιμοποιούνται σε αλγόριθμούς επιτηρούμενης μάθησης αποτελούνται από δεδομένα εισόδου και δεδομένα εξόδου. Από την μία, τα δεδομένα εισόδου είναι τα δεδομένα που μπαίνουν στην είσοδο του αλγορίθμου ενώ τα δεδομένα εξόδου είναι οι τιμές που ο αλγόριθμος πρέπει να προβλέψει για τις αντίστοιχες εισόδους. Χρησιμοποιώντας μαθηματική ορολογία, η επιτηρούμενη μάθηση χρειάζεται σύνολα δεδομένων που περιέχουν τις τιμές **x** και τις τιμές **y** ενός προβλήματος. Ο εκάστοτε αλγόριθμος επιτηρούμενης μάθησης που προσπαθούμε να εκπαιδεύσουμε σε αυτά τα σύνολα δεδομένων υπολογίζει την συνάρτηση **y = f(x),** δηλαδή την συνάρτηση f που αντιστοιχίζει καλύτερα τα δεδομένα **x** στα **y**.

* **Μη επιτηρούμενη μάθηση** (αλλιώς μη επιβλεπόμενη μάθηση ή μάθηση χωρίς επίβλεψη (unsupervised learning)

Χωρίς να παρέχεται κάποια εμπειρία στον αλγόριθμο μάθησης, πρέπει να βρει την δομή των δεδομένων εισόδου [12]. Η μη επιτηρούμενη μάθηση μπορεί να είναι αυτοσκοπός (ανακαλύπτοντας κρυμμένα μοτίβα σε δεδομένα) ή μέσο για ένα τέλος (χαρακτηριστικό της μάθησης). Με άλλα λόγια, τα σύνολα δεδομένων που χρησιμοποιούν οι αλγόριθμοι μη επιτηρούμενης μάθησης περιλαμβάνουν μόνο x τιμές από τις οποίες γίνεται προσπάθεια εύρεσης μοτίβων.

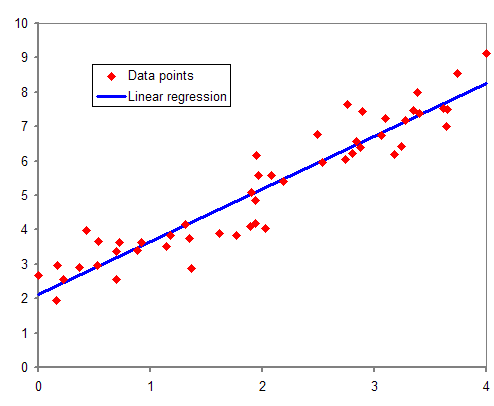
* **Ενισχυτική μάθηση**

Ένα πρόγραμμα υπολογιστή αλληλοεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας συγκεκριμένος στόχος (όπως η οδήγηση ενός οχήματος), χωρίς κάποιος δάσκαλος να του λέει ρητά αν έχει φτάσει κοντά στο στόχο του [12]. Ένα άλλο παράδειγμα είναι να μάθει να παίζει ένα παιχνίδι εναντίον κάποιου αντιπάλου.

#### 2.3.1.2 Με βάση το επιθυμητό αποτέλεσμα του συστήματος μηχανικής μάθησης

* Παλινδρόμηση (Regression)

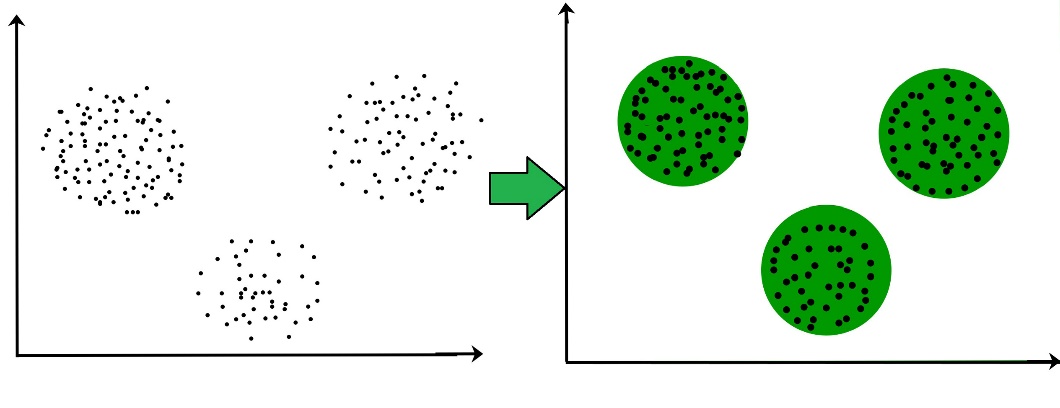
Η παλινδρόμηση αποτελεί πρόβλημα επιτηρούμενης μάθησης όπου η μηχανή πρέπει να κατασκευάσει ένα μοντέλο το οποίο θα προβλέπει συνεχή αποτελέσματα [12]



Εικόνα 23 Παράδειγμα προβλήματος παλινδρόμησης όπου η μπλε γραμμή είναι το μοντέλο που προσπαθεί να προσεγγίσει τις κόκκινες τελείες που αποτελούν συνεχείς τιμές

* Συσταδοποίηση (Clustering)

Η συσταδοποίηση αποτελεί πρόβλημα μη επιτηρούμενης μάθησης όπου η μηχανή καλείται να χωρίσει ένα σύνολο εισόδων σε ομάδες [12]



Εικόνα 2.21 Παράδειγμα συσταδοποίησης όπου σαν είσοδο στο μοντέλο δίνονται τα σημεία του γραφήματος αριστερά και το μοντέλο προσπαθεί τοπολογικά να τα κατηγοριοποιήσει σε 3 συστάδες

* Ταξινόμηση (Classification)

Η ταξινόμηση αποτελεί, κατά βάση, πρόβλημα επιτηρούμενης μάθησης όπου η μηχανή πρέπει να κατασκευάσει ένα μοντέλο το οποίο θα αντιστοιχίζει τα δεδομένα σε δύο ή περισσότερες κλάσεις. [12]

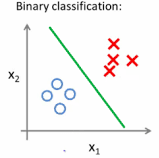
Εικόνα που περιέχει βέλος

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 25 Ταξινόμηση. Η μαύρη γραμμή αντιπροσωπεύει την συνάρτηση f(X,Y) με την οποία ταξινομούνται τα δεδομένα στις κλάσεις Α και Β

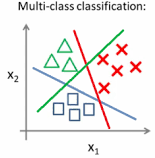
Η ταξινόμηση χωρίζεται στις ακόλουθες κατηγορίες:

1. Δυαδική Ταξινόμηση (Binary Classification): Είναι ένας τύπος ταξινόμησης όπου υπάρχουν δύο διαθέσιμες κλάσεις για την ταξινόμηση των δεδομένων (π.χ. Σωστό ή Λάθος)



Εικόνα 2.22 Δυαδική ταξινόμηση

1. Ταξινόμηση πολλών κλάσεων (Multi-Class Classification): Σε αυτό τον τύπο ταξινόμησης υπάρχουν πάνω από δύο διαθέσιμες κλάσεις για την ταξινόμηση των δεδομένων. Κάθε δείγμα του συνόλου δεδομένων ανατίθεται σε μία κλάση.



Εικόνα 2.23 Ταξινόμηση 3 κλάσεων

1. Ταξινόμηση πολλών κλάσεων (Multi-Label Classification): Σε αυτό τον τύπο ταξινόμησης υπάρχουν πάνω από δύο διαθέσιμες κλάσεις για την ταξινόμηση των δεδομένων. Κάθε δείγμα του συνόλου δεδομένων μπορεί να ανατεθεί περισσότερες από μία κλάσεις. Για παράδειγμα μια ταινία μπορεί να ανήκει σε περισσότερα από ένα είδη (π.χ. περιπέτεια, τρόμου κλπ.)



Εικόνα 2.24 Multi-class vs Multi-label Classification

## 2.4 Σύνολα δεδομένων που χρησιμοποιήθηκαν

Για την παρούσα διπλωματική εργασία χρησιμοποιήθηκαν τρία σύνολα δεδομένων που είναι ελεύθερα στο διαδίκτυο. Στον πίνακα παρουσιάζονται τα σύνολα αυτά μαζί με σημαντικά χαρακτηριστικά αυτών.

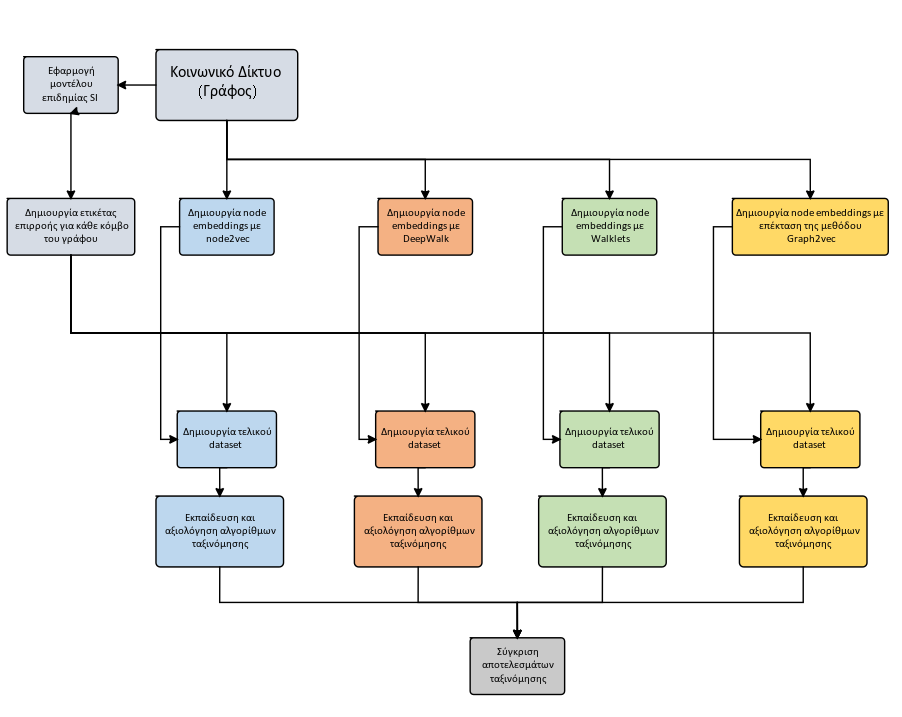
|  |  |  |  |
| --- | --- | --- | --- |
| Δίκτυο | Τύπος Δικτύου | Σύνολο Κόμβων | Σύνολο ακμών |
| CollegeMsg | Κατευθυνόμενος | 1899 | 20296 |
| Facebook | Μη κατευθυνόμενος | 4039 | 88234 |
| Wiki-Vote | Κατευθυνόμενος | 7115 | 103689 |

Πίνακας 2.1 Σύνολα δεδομένων

1. **CollegeMsg** [13]: Αυτό το σύνολο δεδομένων αποτελείται από ιδιωτικά μηνύματα που στάλθηκαν σε ένα μέσο κοινωνικής δικτύωσης στο Πανεπιστήμιο της California. Σε αυτό το μέσο, οι χρήστες μπορούν να αναζητήσουν άλλους χρήστες και να ξεκινήσουν μία συνομιλία με βάση τις πληροφορίες του προφίλ. Μία ακμή (u,v,t) σημαίνει ότι ο χρήστης u έστειλε ένα ιδιωτικό μήνυμα στον χρήστη v την χρονική στιγμή t [14]. Για τις ανάγκες τις παρούσας εργασίας από το δίκτυο αφαιρέθηκε η στήλη t και το δίκτυο εξετάστηκε ως ένας στατικός κατευθυνόμενος γράφος.
2. **Facebook** [15]: Αυτό το σύνολο δεδομένων αποτελείται από κοινωνικούς κύκλους δηλαδή λίστες φίλων από το Facebook. Τα δεδομένα συλλέχθηκαν από μετέχοντες σε έρευνα που χρησιμοποιούν την εφαρμογή του Facebook. Το σύνολο δεδομένων περιλαμβάνει χαρακτηριστικά κόμβου (προφίλ), κύκλους και ego δίκτυα. Τα δεδομένα του Facebook έχουν γίνει ανώνυμα μέσω αντικατάστασης των εσωτερικών αναγνωριστικών του Facebook για κάθε χρήστη με μια νέα τιμή. Επίσης, ενώ έχουν παρασχεθεί διανύσματα χαρακτηριστικών από αυτό το σύνολο δεδομένων, η ερμηνεία αυτών των χαρακτηριστικών έχει αποκρυφθεί. Για παράδειγμα, όπου το αρχικό σύνολο δεδομένων μπορεί να περιείχε ένα χαρακτηριστικό "πολιτικό = Δημοκρατικό Κόμμα", τα νέα δεδομένα θα περιείχαν απλώς "πολιτικό = ανώνυμο χαρακτηριστικό 1". Έτσι, χρησιμοποιώντας τα ανωνυμοποιημένα δεδομένα είναι δυνατόν να προσδιοριστεί εάν δύο χρήστες έχουν τις ίδιες πολιτικές πεποιθήσεις, αλλά όχι τι αντιπροσωπεύουν οι ατομικές πολιτικές τους πεποιθήσεις. [16]
3. **Wiki-Vote** [17]**:** Η Wikipedia είναι μια ελεύθερη γραπτή εγκυκλοπαίδεια, στην οποία μπορούν να συνεισφέρουν εθελοντές από όλο τον κόσμο. Ένα μικρό μέρος των συνεργατών της Wikipedia είναι διαχειριστές, οι οποίοι είναι χρήστες με πρόσβαση σε πρόσθετα τεχνικά χαρακτηριστικά που βοηθούν στη συντήρηση. Προκειμένου να γίνει κάποιος χρήστης διαχειριστής, εκδίδεται μία αίτηση και η κοινότητα της Wikipedia μέσω δημόσιας συζήτησης ή ψηφοφορίας αποφασίζει ποιος θα προωθηθεί για τη διαχείριση. Το παρόν δίκτυο περιέχει όλα τα δεδομένα ψηφοφορίας της Wikipedia από την έναρξη της Wikipedia έως τον Ιανουάριο του 2008. Οι κόμβοι στο δίκτυο αντιπροσωπεύουν τους χρήστες της Wikipedia και μια κατευθυνόμενη ακμή από τον κόμβο i έως τον κόμβο j αντιπροσωπεύει τον χρήστη i που ψήφισε το χρήστη j. [18]

# Κεφάλαιο 3: Μεθοδολογία

Σε αυτό το κεφάλαιο, παρουσιάζεται με τρόπο λεππτομερή η μεθοδολογία που ακολουθήθηκε για την πρόβλεψη της επιρροής των κόμβων σε κοινωνικά δίκτυα με την χρήση embeddings. Το πρόβλημα προσεγγίστηκε από την σκοπιά της ταξινόμησης των κόμβων. Η μεθοδολογία χωρίζεται σε πέντε ενότητες. Αρχικά γίνεται η αξιολόγηση της επιρροής των κόμβων με την βοήθεια του μοντέλου επιδημίας SI. Από αυτή την αξιολόγηση προκύπτουν ετικέτες που δείχνουν την πραγματική σημασία κάθε κόμβου. Στην συνέχεια γίνεται η δημιουργία των node embeddings με τέσσερις διαφορετικές μεθόδους απεικόνισης κόμβων σε χώρους με λιγότερες διαστάσεις. Μετά την δημιουργία των node embeddings πραγματοποιείται ο τελικός σχηματισμός του συνόλου δεδομένων εκπαίδευσης-αξιολόγησης τα οποία θα χρησιμοποιηθούν από τα μοντέλα ταξινόμησης. Ακολουθεί η παρουσίαση και η εκπαίδευση των μοντέλων ταξινόμησης που χρησιμοποιήθηκαν και τέλος παρουσιάζεται η διαδικασία αξιολόγησης των αποτελεσμάτων που ακολουθήθηκε.



Εικόνα 3.1 Διάγραμμα ροής μεθοδολογίας που ακολουθήθηκε

## 3.1 Αξιολόγηση επιρροής κόμβων με την βοήθεια του μοντέλου επιδημίας SI

Το πρόβλημα εξετάστηκε με την χρήση διάφορων αλγορίθμων ταξινόμησης της μηχανικής μάθησης όποτε ήταν αναγκαίο να δοθεί σε κάθε κόμβο μία ετικέτα που θα αντικατοπτρίζει την πραγματική επιρροή αυτού στο δίκτυο. Αυτό επετεύχθη με την βοήθεια του μοντέλου επιδημίας SI. Τα μοντέλα επιδημίας είναι μαθηματικά μοντέλα τα οποία χρησιμοποιούνται σε εφαρμογές όπως η μοντελοποίηση μιας επιδημίας ή η διάδοση μιας ιδέας-άποψης σε ένα κοινωνικό δίκτυο. Στην παρούσα διπλωματική εργασία χρησιμοποιείται το μοντέλο επιδημίας Susceptible-Infected (SI) για να αξιολογηθεί η επιρροή κάθε κόμβου με βάση την συχνότητα συμμετοχής του σε εξαπλώσεις (spreads) που δημιουργούνται σύμφωνα με το SI μοντέλο.

### 3.1.1 Μοντέλο επιδημίας SI

Το επιδημιολογικό μοντέλο (S-I) είναι ένα από τα πιο βασικά μοντέλα επιδημίας και πρωτοσυστήθηκε από τον Kermack (1927). Στο S-I κάθε κόμβος του δικτύου βρίσκεται σε μία από τις δύο καταστάσεις:

• Susceptible (S) (Επιρρεπής): αντιπροσωπεύει τα άτομα του πληθυσμού που είναι υγιή αλλά μπορούν να μολυνθούν.

• Infected (Ι) (Μολυσμένος): υποδηλώνει τα άτομα που έχουν μολυνθεί και είναι σε θέση να μεταδώσουν την ασθένεια στους ευαίσθητους γείτονές τους με πιθανότητα β.Στο μοντέλο αυτό οι κόμβοι μπορούν να μεταφερθούν μόνο από την κατάσταση του επιρρεπή στην κατάσταση του μολυσμένου. Πρακτικά είναι το πιο απλοποιημένο μοντέλο και αυτό στο οποίο έχουν βασιστεί διάφορα άλλα (SIR, SIS, SIRS κ.α.)



Εικόνα 3.2 Το μοντέλο S-I

Οι παραπάνω λοιπόν καταστάσεις, περιγράφονται με τις αντίστοιχες μεταβλητές S(t), I(t) οι οποίες δηλώνουν τον αριθμό των ατόμων σε κάθε κατηγορία σε χρόνο t.

Oι σχέσεις που το περιγράφουν είναι οι ακόλουθες :

St+1 = St – β \* St \* It

It+1 = It + β \* St \* It

Το SI υποθέτει ότι αν, κατά την διάρκεια ενός βήματος της προσομοίωσης, ένας επιρρεπής κόμβος έρθει σε επαφή με ένα μολυσμένο κόμβο, τότε μετατρέπεται σε μολυσμένο με πιθανότητα β.

Οι παράμετροι του μοντέλου είναι:

* beta: Πιθανότητα μόλυνσης
* fraction\_infected: Αρχική ποσότητα μολυσμένων κόμβων

### 3.1.2 Αξιοποίηση μοντέλου SI

Για την εύρεση της πραγματικής επιρροής κάθε χρησιμοποιήθηκε το μοντέλο επιδημίας S-I [19]. Πραγματοποιήθηκαν 500 προσομοιώσεις για κάθε σύνολο δεδομένων που χρησιμοποιήθηκε. Οι τιμές των παραμέτρων των προσομοιώσεων SI ορίστηκαν ως

* beta = 0.1
* fraction infected = 0.05

και ο αριθμός των βημάτων του αλγορίθμου ορίστηκε ως n=3. Για κάθε προσομοίωση του μοντέλου υπολογίζεται το συνολικό πλήθος (σε απόλυτο αριθμό και σε ποσοστό) των κόμβων που έγιναν Infected καθώς και η πλήρης αλληλουχία κόμβων που έχουν γίνει Infected. Αυτή η αλληλουχία δείχνει την εξάπλωση της πληροφορίας και για αυτό ονομάζεται και spread. Μέρος μίας προσομοίωσης από τις 500 που έγιναν για κάθε σύνολο δεδομένων φαίνεται στην εικόνα.



Εικόνα 3.3 Παράδειγμα ενός από τα πεντακόσια spreads που δημιουργήθηκαν με την βοήθεια του μοντέλου SI όπου το σύνολο των κόμβων που έγιναν Infected είναι 460, το ποσοστό του πληθυσμου που έγινε Infected είναι 24.22% και τέλος μία λιστα που φαίνεται ότι οι κόμβοι '761', '395' κλπ έχουν γίνει Infected

### 3.1.3 Υπολογισμός επιρροής κάθε κόμβου

Για την αξιολόγηση της σημασίας των κόμβων του δικτύου χρησιμοποιήθηκε η αλληλουχία των κόμβων. Στην παρούσα διπλωματική εργασία, ως κριτήριο σημασίας ενός κόμβου χρησιμοποιήθηκε η συχνότητα εμφάνισης του σε ένα τέτοιο spread. Με λίγα λόγια, για κάθε κόμβο του δικτύου υπολογίστηκε η τιμή

όπου x = 500. Άρα για κάθε κόμβο υπολογίζεται μία τιμή από 0 έως 1 όπου όσο μεγαλύτερη είναι αυτή, τόσο μεγαλύτερη σημασία έχει ο κόμβος για το δίκτυο. Για την τοποθέτηση των ετικετών επιρροής σε κάθε κόμβο, είναι απαραίτητη η συσταδοποίηση των κόμβων με βάση την τιμή αυτή. Για να γίνει αυτή η συσταδοποίηση χρησιμοποιήθηκε ο αλγόριθμος φυσικών ορίων Jenks.

### 3.1.4 Αλγόριθμος φυσικών ορίων του Jenks

Ο συγκεκριμένος αλγόριθμος βελτιστοποίησης των φυσικών ορίων είναι ένας αλγόριθμος συσταδοποίησης δεδομένων για την εύρεση της καλύτερης κατανομής τιμών σε διαφορετικές κλάσεις. Αυτό επιτυγχάνεται μέσω της ελαχιστοποίησης της μέσης απόκλισης κάθε κλάσης από το μέσο της ίδιας της κλάσης και της μεγιστοποίησης της απόκλισης κάθε κλάσης από τις υπόλοιπες κλάσεις. Με λίγα λόγια, αυτή η μέθοδος αναζητά τρόπο για να μειώσει την διακύμανση μέσα σε κάθε κλάση και να την μεγιστοποιήσει μεταξύ των κλάσεων [20]. Η διακύμανση-διασπορά στην στατιστική ορίζεται ως η αναμενόμενη τιμή της τετραγωνικής απόκλισης της τυχαίας μεταβλητής από την μέση τιμή. Ουσιαστικά μετράει πόσο μακριά ένα σύνολο τυχαίων αριθμών απλώνεται από την μέση τιμή του. Αρχικά γίνεται διαίρεση των δεδομένων σε κλάσεις με τρόπο πολλές φορές αυθαίρετο, με τον αριθμό των κλάσεων να δίνεται ως παράμετρος στον αλγόριθμο. Στην συνέχεια επαναλαμβάνονται τα παρακάτω βήματα [21]:

1. Υπολογισμός του αθροίσματος των τετραγώνων των αποκλίσεων από το μέσο του συνόλου δεδομένων (SDAM sum of squared deviations for array mean).
2. Υπολογισμός του αθροίσματος των τετραγώνων των αποκλίσεων από το μέσο της κλάσης για κάθε συνδυασμό διαστημάτων και εύρεση του μικρότερου (SCDM sum of squared deviations for class means).
3. Υπολογισμός της μετρικής gvf (goodness of variance fit) που ορίζεται ως

και είναι στο διάστημα από 0 έως 1 όπου 1 το τέλειο ταίριασμα και 0 το χειρότερο.

### 3.1.5 Υπολογισμός ετικετών

Μετά τον υπολογισμό της επιρροής κάθε κόμβου, χρησιμοποιήθηκε ο αλγόριθμος φυσικών ορίων του Jenks για να τοποθετηθούν ταμπέλες σε κάθε κόμβο. Στην παρούσα εργασία εξετάστηκαν δύο περιπτώσεις αριθμού κλάσεων όπου στην μία ο αριθμός ορίστηκε για Ν=3 και στην άλλη για Ν=4. Όταν σε έναν κόμβο τοποθετείται η ταμπέλα 0, σημαίνει ότι ο κόμβος έχει μικρή επιρροή στο δίκτυο ενώ όταν έχει Ν=2 ή Ν=3, για τις δύο περιπτώσεις αντίστοιχα, ο κόμβος έχει μεγάλη επιρροή. Ο διαμοιρασμός των δεδομένων φαίνεται στους ακόλουθους πίνακες.

|  |  |  |  |
| --- | --- | --- | --- |
| ΔΙΚΤΥΟ | 0 | 1 | 2 |
| CollegeMsg | 62% | 25% | 13% |
| Facebook | 39% | 28% | 33% |
| Wiki-Votes | 72% | 12% | 16% |

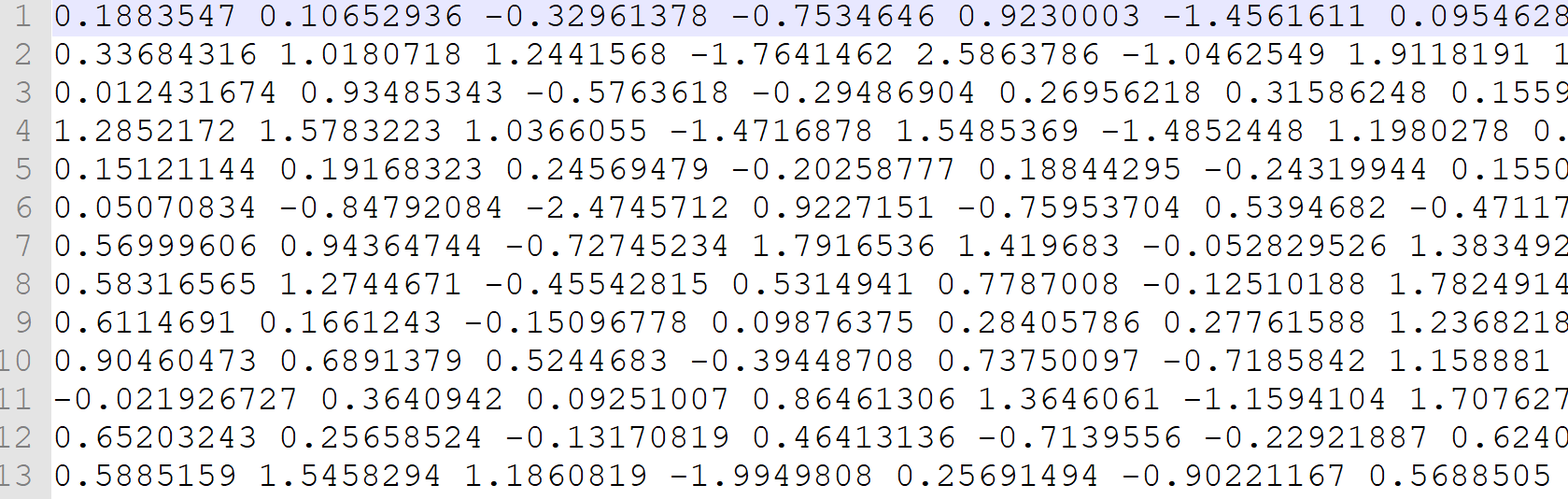
Πίνακας 3.1 Κατανομή παραδειγμάτων ανά κλάση για την περίπτωση των 3 κλάσεων

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ΔΙΚΤΥΟ | 0 | 1 | 2 | 3 |
| CollegeMsg | 53% | 25% | 14% | 8% |
| Facebook | 26% | 24% | 21% | 29% |
| Wiki-Votes | 71% | 9% | 9% | 11% |

Πίνακας 3.2 Κατανομή παραδειγμάτων ανά κλάση για την περίπτωση των 4 κλάσεων

## 3.2 Δημιουργία Node Embeddings

Τα node embeddings είναι ένας τρόπος αναπαράστασης των κόμβων ενός γράφου, σε μορφή διανύσματος αριθμών όπου, για όμοιους κόμβους τα διανύσματα είναι όμοια. Με λίγα λόγια, τα node embeddings δίνουν την δυνατότητα αναπαράστασης κάθε κόμβου σε μορφή τέτοια που να μπορεί να χρησιμοποιηθεί σε εφαρμογές αλγορίθμων μηχανικής μάθησης.



Εικόνα 3.4 Παραδείγματα node embeddings

Στην παρούσα διπλωματική εργασία, που στόχο έχει την πρόβλεψη της ετικέτας επιρροής κάθε κόμβου, τα node embeddings χρησιμοποιήθηκαν για την αναπαράσταση κάθε κόμβου σε μορφή διανύσματος αριθμών έτσι ώστε αυτά τα διανύσματα να μπορούν να αποτελέσουν διανύσματα εισόδου σε μοντέλα ταξινόμησης που έξοδο θα έχουν τις ετικέτες επιρροής κάθε κόμβων. Για την δημιουργία αυτών των αναπαραστάσεων εξετάστηκαν τέσσερις διαφορετικοί αλγόριθμοι με στόχο την τελική σύγκριση της αποτελεσματικότητάς τους στο δεδομένο πρόβλημα ταξινόμησης. Αυτοί είναι οι ακόλουθοι: Node2vec, DeepWalk, Walklets, επέκταση Graph2vec για κόμβους. Για την υλοποίηση τους χρησιμοποιήθηκε η βιβλιοθήκη karateclub [22].

### 3.2.1 Δημιουργία node embeddings με την χρήση της μεθόδου Node2vec

Στο πρώτο πείραμα, για την δημιουργία των node embeddings χρησιμοποιήθηκε ο αλγόριθμος Node2vec. O Node2vec είναι ένας αλγόριθμος δημιουργίας node embeddings που βασίζεται σε τυχαίους περιπάτους. Οι βασικότεροι παράμετροι του μοντέλου αυτού είναι:

* p: Η παράμετρος p ουσιαστικά ελέγχει την πιθανότητα, κατά την διάρκεια του τυχαίου περιπάτου, της άμεσης επιστροφής σε κόμβο που μόλις επισκεφτήκαμε.
* q: Η παράμετρος q ελέγχει πόσο πιθανό είναι, κατά την διάρκεια του τυχαίου περίπατου, να μείνουμε στην γειτονιά του κόμβου u ή να απομακρυνθούμε από αυτήν.
* Διαστάσεις (dimensions): Με την παράμετρο αυτή καθορίζεται το μέγεθος του τελικού διανύσματος απεικόνισης κάθε κόμβου, δηλαδή από πόσους αριθμούς αποτελείται η τελική αναπαράσταση του κάθε κόμβο.
* Αριθμός τυχαίων περιπάτων (number of walks): Με την παράμετρο αυτή δηλώνεται ο αριθμός των τυχαίων περιπάτων που θα γίνονται για κάθε κόμβο.
* Μέγεθος ενός τυχαίου περίπατου (walk length): Με την παράμετρο αυτή δηλώνεται ο αριθμός των κόμβων σε κάθε περίπατο.

Η επιλογή των καλύτερων παραμέτρων για τα node embeddings είναι ένα συζητήσιμο θέμα στον τομέα της έρευνας. Το αντίκτυπο που έχει η μία παράμετρος στις άλλες καθώς και η επίδραση αυτών των παραμέτρων στις επιδόσεις δεν έχουν βρεθεί ακόμα. Στην πράξη, η μέθοδος που χρησιμοποιείται ευρέως για την εύρεση ενός καλού συνόλου παραμέτρων είναι η τυχαία αναζήτηση και δοκιμή. [23]

Στην παρούσα διπλωματική εργασία, για κάθε παράμετρο δημιουργήθηκε ένα σύνολο τιμών. Τα σύνολα τιμών κάθε παραμέτρου είναι τα ακόλουθα.

* p = [0.25 , 0.5 , 1 , 2 , 4]
* q = [0.25 , 0.5 , 1 , 2 , 4]
* Διαστάσεις = [16 , 32 , 64 , 128 , 256]
* Αριθμός τυχαίων περιπάτων = [10 , 15 , 20 , 25]
* Μέγεθος ενός τυχαίου περίπατου = [40 , 60 , 100 , 120]

Στην συνέχεια από τα σύνολα αυτά επελέγησαν 10 διαφορετικοί τυχαίοι συνδυασμοί παραμέτρων και για αυτούς τους συνδυασμούς δημιουργήθηκαν, με την χρήση του αλγόριθμου Νode2vec, 10 διαφορετικά σύνολα node embeddings, για κάθε σύνολο δεδομένων. Οι τιμές των παραμέτρων κάθε συνόλου node embeddings που δημιουργήθηκε με τον αλγόριθμο Node2vec είναι οι ακόλουθες.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| p | q | dimensions | Number of walks | Walk length |
| 0.25 | 0.25 | 64 | 15 | 60 |
| 0.5 | 0.25 | 256 | 10 | 120 |
| 0.25 | 0.25 | 16 | 15 | 60 |
| 4 | 4 | 16 | 15 | 120 |
| 0.5 | 2 | 256 | 20 | 40 |
| 0.5 | 2 | 64 | 20 | 60 |
| 0.5 | 1 | 16 | 10 | 120 |
| 0.25 | 1 | 64 | 20 | 40 |
| 2 | 4 | 16 | 25 | 40 |
| 4 | 1 | 256 | 20 | 60 |

Πίνακας 3.3 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου Node2vec

### 3.2.2 Δημιουργία node embeddings με την χρήση της μεθόδου DeepWalk

Στο δεύτερο πείραμα, για την δημιουργία των node embeddings χρησιμοποιήθηκε ο αλγόριθμος DeepWalk. O DeepWalk είναι ένας αλγόριθμος δημιουργίας node embeddings που βασίζεται σε τυχαίους περιπάτους. Οι τυχαίοι περίπατοι του DeepWalk είναι εντελώς τυχαίοι σε αντίθεση με τον Node2vec όπου οι τυχαίοι περίπατοι είναι προκατειλημμένοι μέσω των παραμέτρων p και q. Οι βασικότεροι παράμετροι του μοντέλου αυτού είναι:

* Διαστάσεις (dimensions): Με την παράμετρο αυτή καθορίζεται το μέγεθος του τελικού διανύσματος απεικόνισης κάθε κόμβου, δηλαδή από πόσους αριθμούς αποτελείται η τελική αναπαράσταση του κάθε κόμβο.
* Αριθμός τυχαίων περιπάτων (number of walks): Με την παράμετρο αυτή δηλώνεται ο αριθμός των τυχαίων περιπάτων που θα γίνονται για κάθε κόμβο.
* Μέγεθος ενός τυχαίου περίπατου (walk length): Με την παράμετρο αυτή δηλώνεται ο αριθμός των κόμβων σε κάθε περίπατο.

Σε αυτή την περίπτωση όπως ακριβώς και στην περίπτωση του Νode2vec, για κάθε παράμετρο δημιουργήθηκε ένα σύνολο τιμών. Τα σύνολα τιμών κάθε παραμέτρου είναι τα ακόλουθα.

* Διαστάσεις = [16 , 32 , 64 , 128 , 256]
* Αριθμός τυχαίων περιπάτων = [10 , 15 , 20 , 25]
* Μέγεθος ενός τυχαίου περίπατου = [40 , 60 , 100 , 120]

Στην συνέχεια, ακολουθώντας την ίδια διαδικασία με την περίπτωση του Νode2vec, από τα σύνολα αυτά επελέγησαν 10 διαφορετικοί τυχαίοι συνδυασμοί παραμέτρων και για αυτούς τους συνδυασμούς δημιουργήθηκαν, με την χρήση του αλγόριθμου DeepWalk, 10 διαφορετικά σύνολα node embeddings, για κάθε σύνολο δεδομένων. Οι τιμές των παραμέτρων κάθε συνόλου node embeddings που δημιουργήθηκε με τον αλγόριθμο DeepWalk είναι οι ακόλουθες.

|  |  |  |
| --- | --- | --- |
| dimensions | Number of walks | Walk length |
| 16 | 25 | 60 |
| 16 | 25 | 100 |
| 32 | 10 | 100 |
| 32 | 10 | 60 |
| 64 | 25 | 100 |
| 64 | 20 | 40 |
| 128 | 20 | 120 |
| 128 | 20 | 100 |
| 256 | 15 | 40 |
| 256 | 25 | 120 |

Πίνακας 3.4 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου DeepWalk

### 3.2.3 Δημιουργία node embeddings με την χρήση της μεθόδου Walklets

Στο τρίτο πείραμα, για την δημιουργία των node embeddings χρησιμοποιήθηκε ο αλγόριθμος Walklets. O Walklets είναι ένας αλγόριθμος δημιουργίας node embeddings που βασίζεται σε τυχαίους περιπάτους. Αποτελεί παραλλαγή του DeepWalk με την διαφορά τους να έγκειται στο ότι μερικοί κόμβοι, κατά την διάρκεια των τυχαίων περιπάτων προσπερνιόνται. Οι βασικότεροι παράμετροι του μοντέλου αυτού είναι:

* Διαστάσεις (dimensions): Με την παράμετρο αυτή καθορίζεται το μέγεθος του τελικού διανύσματος απεικόνισης κάθε κόμβου, δηλαδή από πόσους αριθμούς αποτελείται η τελική αναπαράσταση του κάθε κόμβο.
* Αριθμός τυχαίων περιπάτων (number of walks): Με την παράμετρο αυτή δηλώνεται ο αριθμός των τυχαίων περιπάτων που θα γίνονται για κάθε κόμβο.
* Μέγεθος ενός τυχαίου περίπατου (walk length): Με την παράμετρο αυτή δηλώνεται ο αριθμός των κόμβων σε κάθε περίπατο.

Σε αυτή την περίπτωση όπως ακριβώς και στις προηγούμενες περιπτώσεις, για κάθε παράμετρο δημιουργήθηκε ένα σύνολο τιμών. Τα σύνολα τιμών κάθε παραμέτρου είναι τα ακόλουθα.

* Διαστάσεις = [16 , 32 , 64 , 128 , 256]
* Αριθμός τυχαίων περιπάτων = [10 , 15 , 20 , 25]
* Μέγεθος ενός τυχαίου περίπατου = [40 , 60 , 100 , 120]

Στην συνέχεια, ακολουθώντας την ίδια διαδικασία τις προηγούμενες περιπτώσεις, από τα σύνολα αυτά επελέγησαν 10 διαφορετικοί τυχαίοι συνδυασμοί παραμέτρων και για αυτούς τους συνδυασμούς δημιουργήθηκαν, με την χρήση του αλγόριθμου Walklets, 10 διαφορετικά σύνολα node embeddings. Τα τυχαία σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία των node embeddings είναι τα ακόλουθα.

|  |  |  |
| --- | --- | --- |
| dimensions | Number of walks | Walk length |
| 16 | 25 | 60 |
| 16 | 25 | 100 |
| 32 | 10 | 100 |
| 32 | 10 | 60 |
| 64 | 25 | 100 |
| 64 | 20 | 40 |
| 128 | 20 | 120 |
| 128 | 20 | 100 |
| 256 | 15 | 40 |
| 256 | 25 | 120 |

Πίνακας 3.5 Σύνολα παραμέτρων που χρησιμοποιήθηκαν για την δημιουργία node embeddings μέσω του αλγορίθμου Walklets

### 3.2.4 Δημιουργία node embeddings με την χρήση μιας επέκτασης της μεθόδου Graph2vec για κόμβους (ext-Graph2vec)

Στο τέταρτο πείραμα, για την δημιουργία των node embeddings χρησιμοποιήθηκε μια επέκταση του αλγόριθμου Graph2vec για κόμβους. Σε αντίθεση με τους αλγόριθμους των άλλων περιπτώσεων, που χρησιμοποιούνται για την δημιουργία embeddings για κάθε κόμβο, o Graph2vec είναι ένας αλγόριθμος δημιουργίας embeddings για ολόκληρους γράφους, δηλαδή μετατρέπει έναν γράφο σε ένα διάνυσμα. Για να δημιουργηθούν αναπαραστάσεις σε επίπεδο κόμβου ακολουθήθηκε η εξής διαδικασία.

1. Για κάθε κόμβο, σχηματίζεται ένας υπογράφος που αποτελείται από τις ακμές του γράφου στις οποίες συμμετέχει. Στην περίπτωση των κατευθυνόμενων γράφων, ο υπογράφος κάθε κόμβου σχηματίζεται από όλες τις ακμές στις οποίες περιλαμβάνεται (είτε αυτές εκκινούν από τον κόμβο είτε καταλήγουν σε αυτόν). Άρα για γράφο με Ν κόμβους, προκύπτουν Ν υπογράφοι.
2. Για κάθε υπογράφο που δημιουργήθηκε στο πρώτο βήμα εφαρμόζεται ο αλγόριθμος Graph2vec και δημιουργείται ένα embedding για κάθε υπογράφο, δηλαδή για κάθε κόμβο.

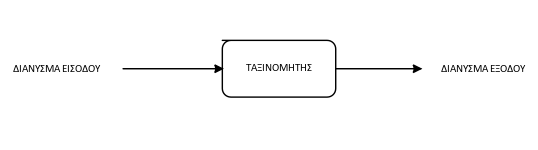
Η βασική παράμετρος του αλγόριθμου Graph2vec και της κατ’ επέκταση της νέας αυτής τεχνικής είναι διαστάσεις των διανυσμάτων. Όπως και στις προηγούμενες περιπτώσεις δημιουργήθηκε ένα σύνολο τιμών για την παράμετρο αυτή που είναι το

* Διαστάσεις = [16 , 32 , 64 , 128 , 256].

Σε αυτή την περίπτωση δημιουργήθηκαν node embeddings για όλες τις τιμές του συνόλου τιμών (και για τις 5 τιμές).

## 3.3 Σχηματισμός τελικού συνόλου δεδομένων εκπαίδευσης-αξιολόγησης

Για να είναι δυνατή η υλοποίηση των ταξινομητών που στόχο έχουν την κατηγοριοποίηση των κόμβων με βάση την επιρροή τους στο δίκτυο, πρέπει να σχηματιστεί η τελική μορφή των δεδομένων εκπαίδευσης και αξιολόγησης. Στην είσοδο κάθε ταξινομητή μπαίνει ένα διάνυσμα που λέγεται διάνυσμα εισόδου και στην έξοδο η κατηγορία στην οποία ταξινομείται το συγκεκριμένο διάνυσμα.



Εικόνα 3.5 Δομή ταξινομητή

Ενώ η έξοδος που θα χρησιμοποιηθεί στον ταξινομητή, δηλαδή η κατηγορία στην οποία ανήκει κάθε κόμβος με βάση την επιρροή του, υπολογίστηκε, δεν διευκρινίστηκαν τα διανύσματα εισόδου που θα χρησιμοποιηθούν. Για διανύσματα εισόδου δημιουργήθηκαν node embeddings με την χρήση τεσσάρων διαφορετικών μεθόδων, όπου για κάθε μέθοδο χρησιμοποιήθηκε ένα σύνολο από τυχαίους συνδυασμούς τιμών για τις παραμέτρους αυτής. Για να καταλήξουμε στο τελικό σύνολο node embeddings που θα χρησιμοποιηθεί, έπρεπε με κάποιο τρόπο να επιλεγεί για κάθε πείραμα το σύνολο που θα δώσει το καλύτερο αποτέλεσμα ταξινόμησης. Κατόπιν πολλών δοκιμών ταξινόμησης χρησιμοποιώντας τον αλγόριθμο λογιστικής παλινδρόμησης σε συνδυασμό με την μέθοδο K-Fold cross validation των συνόλων δεδομένων, τα σύνολα παραμέτρων που έδωσαν τα καλύτερα αποτελέσματα φαίνονται στον παρακάτω πίνακα:

|  |  |  |  |
| --- | --- | --- | --- |
|  | CollegeMsg | Facebook | Wiki-Vote |
| Node2vec | p = 0.25  q = 1  dimensions = 64  number of walks = 20  walk length = 40 | p = 0.5  q = 0.25  dimensions = 256  number of walks = 10  walk length = 120 | p = 0. 5  q = 2  dimensions = 256  number of walks = 20  walk length = 40 |
| DeepWalk | dimensions = 256  number of walks = 15  walk length = 40 | dimensions = 120  number of walks = 20  walk length = 120 | dimensions = 120  number of walks = 20  walk length = 120 |
| Walklets | dimensions = 32  number of walks = 10  walk length = 60 | dimensions = 128  number of walks = 20  walk length = 100 | dimensions = 64  number of walks = 25  walk length = 100 |
| ext-Graph2vec | dimensions = 16 | dimensions = 256 | dimensions = 128 |

Πίνακας 3.6 Παράμετροι node embeddings για κάθε dataset για κάθε μέθοδο δημιουργίας node embeddings (3 κλάσεις)

|  |  |  |  |
| --- | --- | --- | --- |
|  | CollegeMsg | Facebook | Wiki-Vote |
| Node2vec | p = 0.25  q = 0.25  dimensions = 64  number of walks = 15  walk length = 60 | p = 0.5  q = 0.25  dimensions = 256  number of walks = 10  walk length = 120 | p = 0. 5  q = 2  dimensions = 256  number of walks = 20  walk length = 40 |
| DeepWalk | dimensions = 16  number of walks = 25  walk length = 100 | dimensions = 128  number of walks = 20  walk length = 100 | dimensions = 256  number of walks = 15  walk length = 40 |
| Walklets | dimensions = 256  number of walks = 15  walk length = 40 | dimensions = 128  number of walks = 20  walk length = 120 | dimensions = 64  number of walks = 25  walk length = 100 |
| ext-Graph2vec | dimensions = 32 | dimensions = 256 | dimensions = 128 |

Πίνακας 3.7 Παράμετροι node embeddings για κάθε dataset dataset για κάθε μέθοδο δημιουργίας node embeddings (4 κλάσεις)

Τέλος, σε μερικές περιπτώσεις όπου υπήρχε μεγάλη ανισορροπία δεδομένων μεταξύ κλάσεων (δηλαδή σε κάποιες κλάσεις υπήρχαν πολύ περισσότερα παραδείγματα από ότι σε άλλες), κατόπιν δοκιμών, υπήρξε υποδειγματοληψία δεδομένων των κλάσεων αυτών για βελτίωση των αποτελεσμάτων. Με τον όρο υποδειγματοληψία εννοούμε την αφαίρεση παραδειγμάτων από τις κλάσεις που υπερισχύουν αριθμητικά.

## 3.4. Εκπαίδευση μοντέλων μηχανικής μάθησης

Μετά την κατασκευή των τελικών συνόλων δεδομένων εκπαίδευσης και αξιολόγησης ακολουθεί η διαδικασία της ταξινόμησης. Οι αλγόριθμοι ταξινόμησης που χρησιμοποιήθηκαν για το συγκεκριμένο πρόβλημα ταξινόμησης είναι οι ακόλουθοι: α)Naïve Bayes, β)Δέντρα απόφασης (Decision Trees), γ)Κ-πλησιέστεροι γείτονες (K-nearest neighbors), δ)Λογιστική Παλινδρόμηση (Logistic Regression), ε)Μηχανές Διανυσμάτων Υποστήριξης ΜΔΥ (Support Vector Machines, SVM), στ)Νευρωνικό δίκτυο (Neural Network). Τα πέντε πρώτα μοντέλα μηχανικής μάθησης χρησιμοποιήθηκαν από την βιβλιοθήκη scikit-learn ενώ το Νευρωνικό δίκτυο χρησιμοποιήθηκε από την βιβλιοθήκη tensorflow.keras.

### 3.4.1 Gaussian Naive Bayes

Ο Gaussian Naive Bayes [24] είναι ένας πιθανοτικός κατηγοριοποιητής που βασίζεται στο θεώρημα του Bayes, κάνοντας την υπόθεση ότι τα χαρακτηριστικά εισόδου είναι ανεξάρτητα μεταξύ τους. Το θεώρημα του Bayes ορίζεται από την ακόλουθη σχέση

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

όπου, y η μεταβλητή της κλάσης και x = [x1,…,xn] το διάνυσμα χαρακτηριστικών εισόδου. Χρησιμοποιώντας την υπόθεση της ανεξαρτησίας

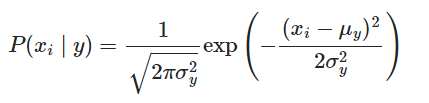


για κάθε i, προκύπτει ο τελικός τύπος:

Εικόνα που περιέχει κείμενο

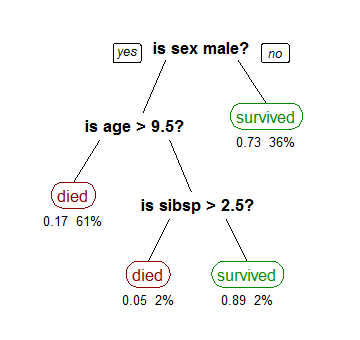
Περιγραφή που δημιουργήθηκε αυτόματα

Η πιθανότητα των χαρακτηριστικών υποτίθεται ότι είναι γκαουσσιανή δηλαδή δίνεται από την σχέση:



### 3.4.2 Δέντρα απόφασης (Decision Tree)

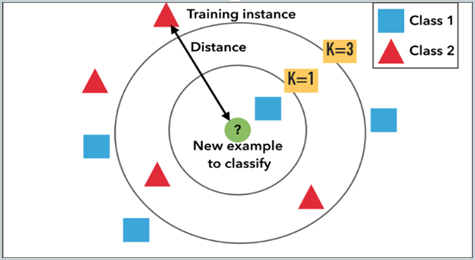
Τα δέντρα απόφασης [25] είναι μία μέθοδος επιβλεπόμενης μάθησης που χρησιμοποιείται για ταξινόμηση (classification) και παρεμβολή (regression). Στόχος είναι η δημιουργία ενός μοντέλου που θα προβλέπει την τιμή της μεταβλητής εξόδου μαθαίνοντας από απλούς κανόνες απόφασης που προκύπτουν από τις μεταβλητές εισόδου.



Εικόνα 3.6 Παράδειγμα ταξινόμησης με την χρήση δέντρου απόφασης

### 3.4.3 K-κοντινότεροι γείτονες (k-nearest neighbors)

Οι K-κοντινότεροι γείτονες [26] είναι μία μέθοδος επιβλεπόμενης μάθησης που χρησιμοποιείται για ταξινόμηση (classification) και παρεμβολή (regression). Η βασική αρχή πίσω από αυτή την μέθοδο είναι η εύρεση ενός προκαθορισμένου αριθμού δειγμάτων που βρίσκονται πιο κοντά (δηλαδή έχουν την μικρότερη απόσταση) σε ένα καινούριο σημείο και η πρόβλεψη της κλάσης από αυτά. Ο αριθμός των δειγμάτων (K) προκαθορίζεται από τον χρήστη.



Εικόνα 3.7 k-nearest neighbors

### 3.4.4 Λογιστική Παλινδρόμηση (Logistic Regression)

Η Logistic Regression [27] είναι μια μέθοδος επιβλεπόμενης μάθησης που χρησιμοποιείται για προβλήματα ταξινόμησης και βασίζεται στην έννοια της πιθανότητας. Χρησιμοποιείται κατά κύριο λόγο σε προβλήματα δυαδικής ταξινόμησης (binary classification) όπου οι πιθανές κατηγορίες στις όποιες ταξινομούνται τα δεδομένα έχουν την μορφή Ναι ή Όχι, 0 ή 1 κλπ. Ουσιαστικά γίνεται πρόβλεψη της πιθανότητας κάθε μιας από τις δύο κατηγορίες.Η καμπύλη της πρόβλεψης έχει σιγμοειδή-λογιστική μορφή

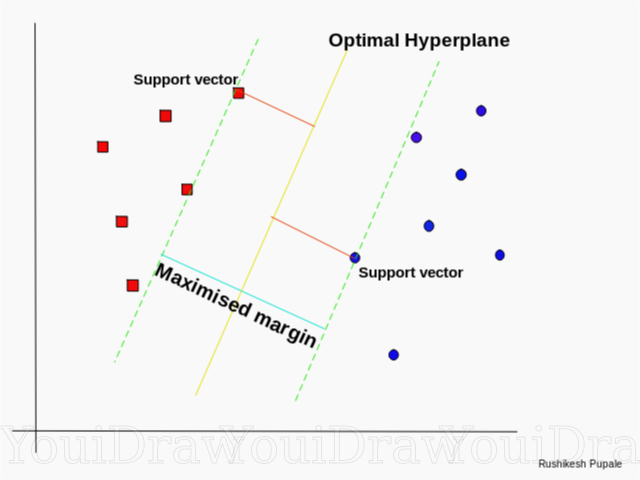
Εικόνα που περιέχει κείμενο, ρολόι

Περιγραφή που δημιουργήθηκε αυτόματα

Στην περίπτωση της ταξινόμησης σε περισσότερες από 2 κατηγορίες χρησιμοποιείται η λογική one-vs-rest όπου το πρόβλημα διαιρείται σε N υποπροβλήματα όπου Ν ο αριθμός των κλάσεων.

### 3.4.5 Μηχανές Διανυσμάτων Υποστήριξης ΜΔΥ (Support Vector Machines, SVM)

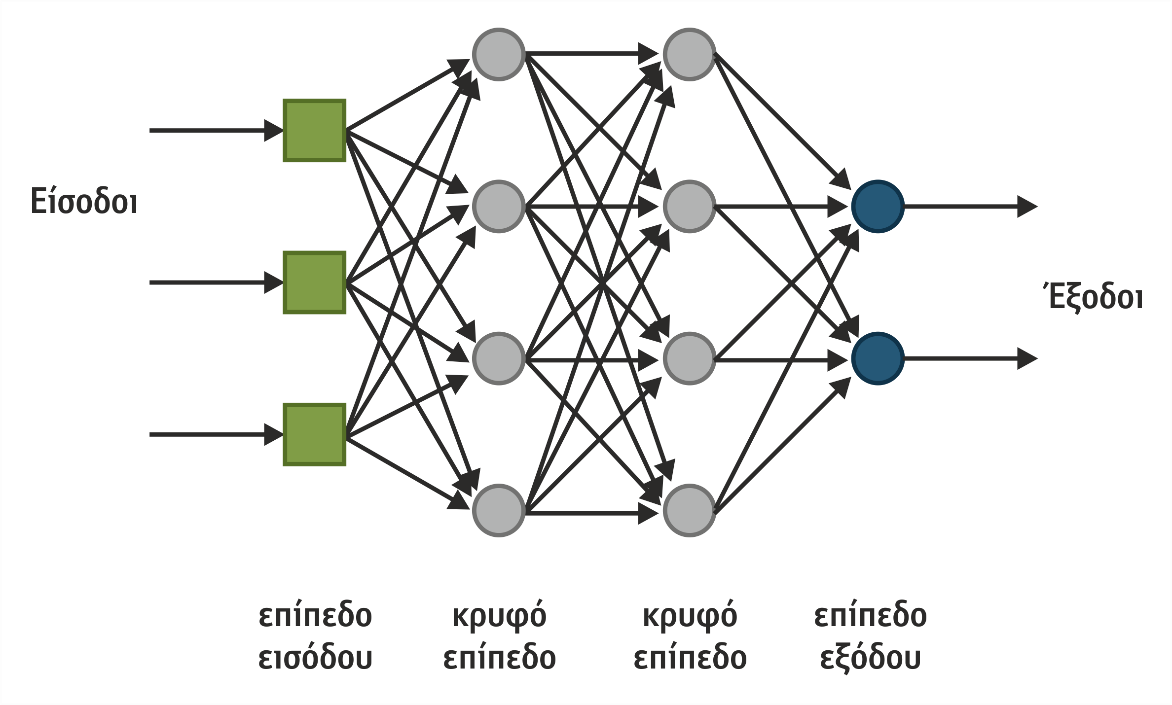
Τα SVM [28] είναι μια οικογένεια αλγορίθμων επιβλεπόμενης μάθησης που χρησιμοποιούνται για προβλήματα ταξινόμησης και παρεμβολής. Βασίζεται στην ιδέα ότι ο αλγόριθμος δημιουργεί μια γραμμή ή ένα hyperplane που χωρίζει τα δεδομένα σε κλάσεις. Για να υπολογιστεί αυτή η γραμμή, αρχικά βρίσκεται ένα σημείο για κάθε κλάση που είναι πιο κοντά στην γραμμή. Αυτά τα σημεία λέγονται διανύσματα υποστήριξης. Στην συνέχεια υπολογίζεται η απόσταση των διανύσματων αυτών από την διαχωριστική γραμμή (ή hyperplane) και τέλος γίνεται προσπάθεια μείωσης αυτής της απόστασης. Το hyperplane σε ένα χώρο ν-διαστάσεων, είναι ένα χώρος ν-1 διαστάσεων που χωρίζει τον χώρο των ν-διαστάσεων σε δύο μέρη.



Εικόνα 3.8 Svm

### 3.4.6 Νευρωνικό δίκτυο

Το νευρωνικό δίκτυο [29] είναι ένα δίκτυο από απλούς υπολογιστικούς κόμβους (νευρώνες, νευρώνια), διασυνδεδεμένους μεταξύ τους. Είναι εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα (ΚΝΣ), το οποίο προσπαθεί να προσομοιώσει. Οι νευρώνες είναι τα δομικά στοιχεία του δικτύου. Κάθε τέτοιος κόμβος δέχεται ένα σύνολο αριθμητικών εισόδων από διαφορετικές πηγές (είτε από άλλους νευρώνες, είτε από το περιβάλλον), επιτελεί έναν υπολογισμό με βάση αυτές τις εισόδους και παράγει μία έξοδο. Η εν λόγω έξοδος είτε κατευθύνεται στο περιβάλλον, είτε τροφοδοτείται ως είσοδος σε άλλους νευρώνες του δικτύου. Υπάρχουν τρεις τύποι νευρώνων: οι νευρώνες εισόδου, οι νευρώνες εξόδου και οι υπολογιστικοί νευρώνες ή κρυμμένοι νευρώνες. Οι νευρώνες εισόδου δεν επιτελούν κανέναν υπολογισμό, μεσολαβούν απλώς ανάμεσα στις περιβαλλοντικές εισόδους του δικτύου και στους υπολογιστικούς νευρώνες. Οι νευρώνες εξόδου διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου. Οι υπολογιστικοί νευρώνες πολλαπλασιάζουν κάθε είσοδό τους με το αντίστοιχο συναπτικό βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων. Το άθροισμα αυτό τροφοδοτείται ως όρισμα στη συνάρτηση ενεργοποίησης, την οποία υλοποιεί εσωτερικά κάθε κόμβος. Η τιμή που λαμβάνει η συνάρτηση για το εν λόγω όρισμα είναι και η έξοδος του νευρώνα για τις τρέχουσες εισόδους και βάρη.



Εικόνα 3.9 Παράδειγμα νευρωνικού δικτύου

Εισάγοντας στο νευρωνικό δίκτυο τα δεδομένα εισόδου και εξόδου, αυτό προσπαθεί να εκπαιδεύσει τα βάρη που ενώνουν τους νευρώνες έτσι ώστε να δίνει έξοδο κοντινή σε αυτή της εκπαίδευσης.

Στην παρούσα διπλωματική εργασία, για την εκπαίδευση του Νευρωνικού Δικτύου και την επιλογή των καταλληλότερων παραμέτρων για αυτό χρησιμοποιήθηκε ο αλγόριθμος Hyperband [30].

#### 3.4.6.1 Επιλογή Παραμέτρων νευρωνικού δικτύου με την μέθοδο Hyperband

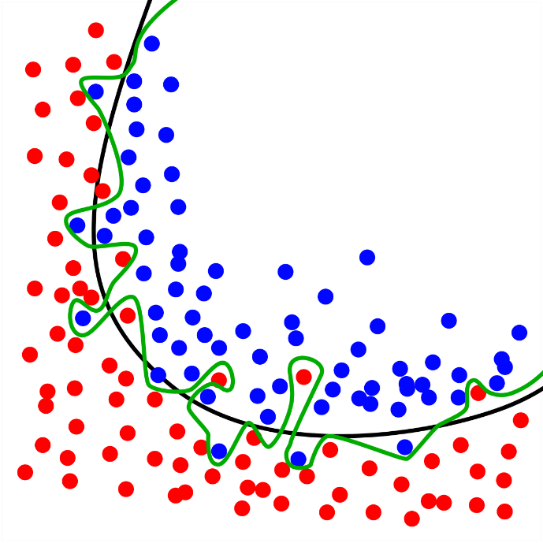
Ο Hyperband [31] είναι ένας αλγόριθμος εύρεσης των καλύτερων παραμέτρων για μοντέλα μηχανικής μάθησης. Όπως οι περισσότεροι τέτοιοι αλγόριθμοι, ουσιαστικά τρέχει επαναληπτικά το μοντέλο για το οποίο προσπαθεί να επιλέξει τιμές για τις παραμέτρους του χρησιμοποιώντας πολλούς διαφορετικούς συνδυασμούς. Επειδή αυτό είναι πολλές φορές χρονοβόρο αντί να τρέχει τον αλγόριθμο, για κάθε συνδυασμό παραμέτρων, μέχρι την ολοκλήρωση του, τον τρέχει για λίγες επαναλήψεις (συνήθως μία με δύο) για να πάρει μια γεύση του πως εκτελείται. Αυτό βασίζεται στην ιδέα ότι για κάποια σύνολα παραμέτρων είναι εμφανές από τις πρώτες επαναλήψεις ότι δεν θα είναι αποτελεσματικά. Στην συνέχεια παίρνει τα σύνολα παραμέτρων που αποδείχτηκαν καλύτερα και τα τρέχει για περισσότερες επαναλήψεις. Ουσιαστικά, αυτό που κάνει το Hyperband είναι να τρέχει το μοντέλο μηχανικής μάθησης για τυχαία σετ παραμέτρων με συγκεκριμένο προγραμματισμό επαναλήψεων για κάθε set, χρησιμοποιώντας πρώιμα αποτελέσματα για να επιλέξει τα υποψήφια σετ παραμέτρων που θα τρέξουν περισσότερες επαναλήψεις.

## 3.5 Αξιολόγηση αποτελεσμάτων

Το σπουδαιότερο πράγμα στην δημιουργία ενός μοντέλου μηχανικής μάθησης είναι η αξιολόγηση της αποτελεσματικότητάς του. Αυτή η αποτελεσματικότητα εξετάζεται με διάφορες μαθηματικές σχέσεις που λέγονται μετρικές αξιολόγησης. Στην παρούσα διπλωματική εργασία για την αξιολόγηση των αποτελεσμάτων των μοντέλων χρησιμοποιήθηκε η μέθοδος K-Fold Cross Validation και ο Πίνακας σύγχυσης (confusion matrix) για τις μετρικές αξιολόγησης: ευστοχία (accuracy), ακρίβεια (precision) , ανάκληση (recall), f1-score.

### 3.5.1 Cross Validation

Το Cross Validation [32] είναι μία μέθοδος επαναδειγματοληψίας η οποία χρησιμοποιεί διαφορετικά τμήματα των δεδομένων για να εκπαιδεύσει και να εξετάσει ένα μοντέλο σε διαφορετικές επαναλήψεις. Χρησιμοποιείται, κατά κύριο λόγο, σε περιπτώσεις που ο στόχος είναι η πρόβλεψη και αποσκοπεί στην εκτίμηση της ακρίβειας που θα παρουσιάσει το μοντέλο στην πράξη. Σε ένα πρόβλημα πρόβλεψης, στην είσοδο του μοντέλου δίνεται ένα σύνολο γνωστών δεδομένων με τα οποία γίνεται η εκπαίδευση (training dataset) και ένα σύνολο μη γνωστών δεδομένων με τα οποία γίνεται ο έλεγχος ακρίβειας του μοντέλου (testing dataset). Ο στόχος του cross validation είναι η εξέταση της ικανότητας που έχει ένα μοντέλο να προβλέπει καινούρια δεδομένα τα οποία δεν έχουν χρησιμοποιηθεί για την εκπαίδευση του. Με αυτό τον τρόπο αντιμετωπίζεται το πρόβλημα της υπερμοντελοποίησης (overfitting) που είναι το υπερβολικό ταίριασμα με τα δεδομένα εκπαίδευσης. Το Overfitting έχει σαν αποτέλεσμα το μοντέλο να έχει ταιριάξει απόλυτα με τα δεδομένα εκπαίδευσης και να μην μπορεί να προβλέψει με ακρίβεια μη γνωστά δεδομένα.



Εικόνα 3.10 Η πράσινη γραμμή αντιπροσωπεύει ένα μοντέλο που κάνει overfitting και η μαύρη ένα κανονικοποιημένο μοντέλο. Όπως βλέπουμε το πράσινο μοντέλο εξαρτάται απόλυτα από τα δεδομένα εκπαίδευσης, πράγμα που σημαίνει ότι πιθανότατα δεν θα έχει καλή ακρίβεια σε δεδομένα που δεν έχει ξαναδεί.

#### 3.5.1.1 K-Fold cross validation

Ο K-Fold [33] είναι η πιο κλασσική προσέγγιση του cross validation. Σε αυτήν, το σύνολο δεδομένων εκπαίδευσης χωρίζεται σε Κ μικρότερα σύνολα που λέγονται folds. Στην συνέχεια το μοντέλο εκπαιδεύεται χρησιμοποιώντας Κ-1 από αυτά τα υποσύνολα ως δεδομένα εκπαίδευσης. Η τελική εκτίμηση για την ακρίβεια πρόβλεψης του μοντέλου προκύπτει από το εναπομένον υποσύνολο το οποίο χρησιμοποιείται ως σύνολο αξιολόγησης.

Εικόνα που περιέχει κείμενο, οθόνη, στιγμιότυπο οθόνης, ασήμι

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.11 Παράδειγμα K-Fold

### 3.5.2 Πίνακας σύγχυσης (Confusion Matrix)

Πολύ χρήσιμο εργαλείο για τον υπολογισμό τέτοιων μετρικών είναι ο πίνακας σύγχυσης (confusion matrix) [34] που είναι ένας ΜxΜ πίνακας, όπου Μ το πλήθος των κλάσεων του προβλήματος ταξινόμησης. Κάθε γραμμή του πίνακα αντιπροσωπεύει τα στιγμιότυπα σε μια προβλεπόμενη κλάση (Predicted), ενώ κάθε στήλη αντιπροσωπεύει τα στιγμιότυπα σε μια πραγματική κλάση (Actual) ή το αντίστροφο - και οι δύο παραλλαγές βρίσκονται στη βιβλιογραφία. Για προβλήματα δυαδικής ταξινόμησης, δηλαδή προβλήματα όπου το σύνολο των κλάσεων είναι 2, ο πίνακας σύγχυσης είναι ένας πίνακας 2x2 όπως φαίνεται στην εικόνα:

Εικόνα που περιέχει πίνακας

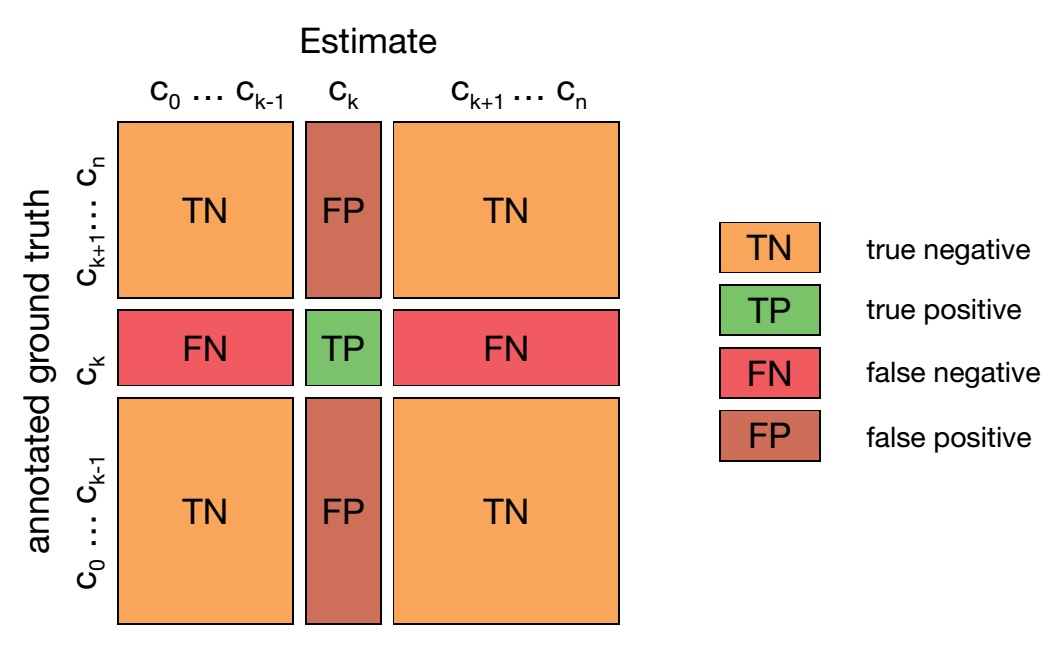
Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 3.12 Πίνακας σύγχυσης για προβλήματα δυαδικής ταξινόμησης

Για τις δύο κλάσεις θετικό-αρνητικό (positive-negative) διακρίνονται συγκεκριμένες περιπτώσεις:

* Αληθώς θετικά (True Positive-TP): Στα αληθώς θετικά τοποθετείται το σύνολο των δειγμάτων που ταξινομήθηκαν ως θετικά και ήταν πράγματι θετικά.
* Αληθώς αρνητικά (True Negative-TN): Στα αληθώς αρνητικά τοποθετείται το σύνολο των δειγμάτων που ταξινομήθηκαν ως αρνητικά και ήταν πράγματι αρνητικά.
* Ψευδώς θετικά (False Positive-FP): Στα ψευδώς θετικά τοποθετείται το σύνολο των δειγμάτων που ταξινομήθηκαν ως θετικά ενώ ήταν αρνητικά.
* Ψευδώς αρνητικά (False Negative-FN): Στα ψευδώς αρνητικά τοποθετείται το σύνολο των δειγμάτων που ταξινομήθηκαν ως αρνητικά ενώ ήταν θετικά.

Σε προβλήματα ταξινόμησης πολλών κλάσεων ουσιαστικά γίνεται δυαδική ταξινόμηση για κάθε κλάση ξεχωριστά για τον υπολογισμό των (TP, TN, FP, FN). Για κάθε κλάση υπολογίζεται μια μετρική αξιολόγησης (πχ ευστοχία), οπότε για Μ κλάσεις θα έχουμε Μ τιμές για την ευστοχία. Για την σύγκριση δύο μοντέλων όμως χρειαζόμαστε μία τιμή για την ευστοχία όποτε χρησιμοποιούμε διάφορες τιμές για την εύρεση μιας μέσης τιμής για το μοντέλο, όπως macro, micro, weighted.



Εικόνα 3.13 Πίνακας σύγχυσης για προβλήματα ταξινόμησης πολλών κλάσεων

* Αληθώς θετικά (True Positive-TP): Η τιμή των αληθώς θετικών για μία κλάση είναι εκεί όπου η πραγματική τιμή με την τιμή που προέκυψε από την πρόβλεψη είναι η ίδια.
* Αληθώς αρνητικά (True Negative-TN): Η τιμή των αληθώς αρνητικών για μία κλάση θα είναι το άθροισμα των στοιχείων όλων των στηλών και όλων των γραμμών, εκτός της γραμμής και της στήλης της κλάσης για την οποία υπολογίζουμε τις τιμές αυτές.
* Ψευδώς θετικά (False Positive-FP): Η τιμή των ψευδώς θετικών για μία κλάση θα είναι το άθροισμα των τιμών της αντίστοιχης στήλης εκτός της τιμής των αληθώς θετικών.
* Ψευδώς αρνητικά (False Negative-FN): Η τιμή των ψευδώς αρνητικών για μία κλάση θα είναι το άθροισμα των τιμών της αντίστοιχης γραμμής εκτός της τιμής των αληθώς θετικών.

### 3.5.3 Μετρικές Ταξινόμησης

Οι μετρικές ταξινόμησης είναι μαθηματικές σχέσεις οι οποίες εξετάζουν την αποτελεσματικότητα των αλγορίθμων μηχανικής μάθησης.

#### 3.5.3.1 Ευστοχία (accuracy)

Η ευστοχία υπολογίζεται από τον λόγο

ενώ με την χρήση των TP, TN, FP, FN δίνεται από την μαθηματική σχέση

#### 3.5.3.2 Ακρίβεια (precision)

Η ακρίβεια είναι το ποσοστό των δειγμάτων που έχουν ταξινομηθεί ως θετικά και είναι πράγματι θετικά

#### 3.5.3.3 Ανάκληση (recall)

Η ανάκληση είναι το ποσοστό των θετικών δειγμάτων που κατάφερε να βρει ο ταξινομητής.

#### 3.5.3.4 F-score

Ο F-score είναι ο αρμονικός μέσος όρος της ακρίβειας και της ανάκλησης

#### 3.5.3.5 Μετρική για σύγκριση μεθόδων

Για την συνολική σύγκριση των μεθόδων ταξινόμησης, υπολογίστηκε η μέση αποτελεσματικότητα που προκύπτει από τον τύπο:

# Κεφάλαιο 4: Αποτελέσματα

Σε αυτό το κεφάλαιο, παρουσιάζονται τα αποτελέσματα τα οποία προέκυψαν κατά την διεξαγωγή των πειραμάτων που πραγματεύεται η παρούσα διπλωματική εργασία. Η κεντρική ιδέα που ακολουθήθηκε κατά την διάρκεια της μεθοδολογίας είναι η πρόβλεψη της επιρροής των κόμβων, μέσω του συνδυασμού αλγορίθμων ταξινόμησης μηχανικής μάθησης και αλγορίθμων απεικόνισης κόμβων σε χαμηλότερες διαστάσεις. Για την απεικόνιση κόμβων σε χαμηλότερες διαστάσεις, δηλαδή την δημιουργία node embeddings, εξετάστηκαν τέσσερις διαφορετικές μέθοδοι δημιουργίας. Πιο συγκεκριμένα, εξετάστηκαν οι

1. Node2vec
2. DeepWalk
3. Walklets
4. Επέκταση αλγορίθμου Graph2vec για κόμβους

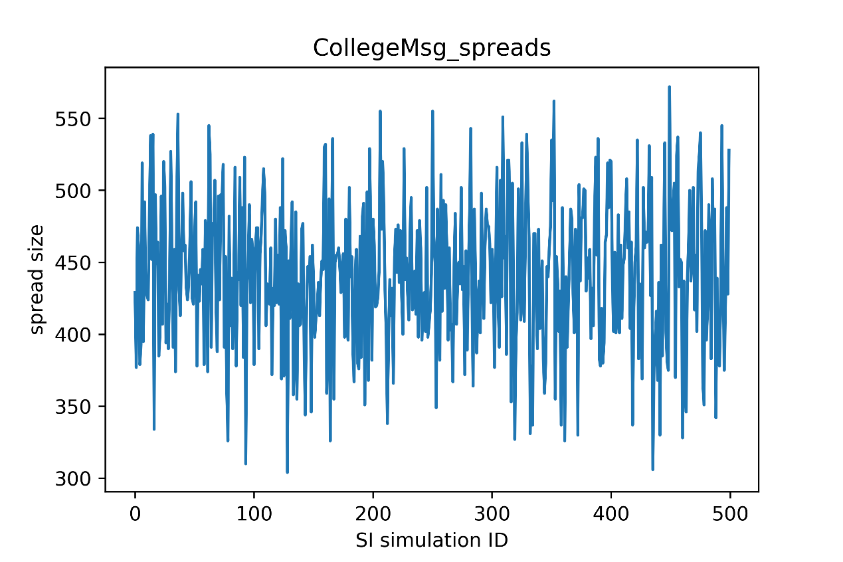
τα αποτελέσματα των οποίων, χωρίζονται σε τέσσερα μέρη αντιστοίχως. Σε κάθε μέθοδο πραγματοποιείται σύγκριση μεταξύ των αποτελεσμάτων κάθε αλγόριθμου ταξινόμησης που χρησιμοποιήθηκε, για κάθε σύνολο δεδομένων. Εκτός της σύγκρισης των αλγορίθμων μηχανικής μάθησης που εξετάζονται, πραγματοποιείται και σύγκριση μεταξύ των αποτελεσμάτων κάθε αλγόριθμου δημιουργίας node embeddings. Με λίγα λόγια, εντοπίζουμε ποιος αλγόριθμος ταξινόμησης βρίσκει με μεγαλύτερη ακρίβεια τους κόμβους επιρροής για κάθε σύνολο δεδομένων καθώς και ποιος συνδυασμός μεταξύ αλγόριθμου ταξινόμησης και αλγόριθμου δημιουργίας node embedding έχει την μεγαλύτερη αποτελεσματικότητα στην ταξινόμηση των κόμβων.

## 4.1 Ταξινόμηση κόμβων συνδυάζοντας Node2vec και μηχανική μάθηση

Σε αυτή την ενότητα προτείνεται η πρώτη μέθοδος για τον χαρακτηρισμό των κόμβων, με βάση την επιρροή που ασκούν στα κοινωνικά δίκτυα. Σύμφωνα με αυτή, ο χαρακτηρισμός των κόμβων αντιμετωπίζεται ως πρόβλημα ταξινόμησης, συνεπώς για τον χαρακτηρισμό των κόμβων χρησιμοποιούνται αλγόριθμοι ταξινόμησης και πιο συγκεκριμένα αλγόριθμοι επιβλεπόμενης ταξινόμησης. Για την δημιουργία δεδομένων εκπαίδευσης, για τα μοντέλα ταξινόμησης, χρησιμοποιήθηκε η μέθοδος Node2vec με την οποία γίνεται απεικόνιση κόμβων σε χώρους με λιγότερες διαστάσεις.

### 4.1.1 Κατασκευή δεδομένων εκπαίδευσης

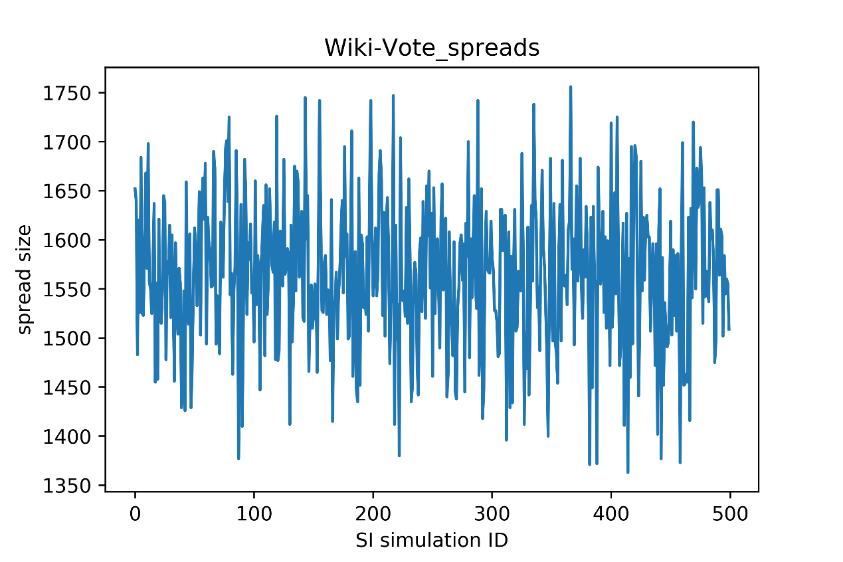
Για την αξιολόγηση της επιρροής κάθε κόμβου με στόχο την δημιουργία ετικετών που είναι αντιπροσωπευτικές αυτής, χρησιμοποιήθηκε το μοντέλο επιδημίας SI και στα 3 σύνολα δεδομένων. Ο αριθμός των προσομοιώσεων που πραγματοποιήθηκε για κάθε σύνολο δεδομένων ορίστηκε σε 500 προσομοιώσεις. Για κάθε προσομοίωση εξετάστηκαν 3 επαναλήψεις του αλγορίθμου, ενώ η πιθανότητα μετάδοσης της ασθένειας (β) και ο αρχικός πληθυσμός των μολυσμένων κόμβων (fraction\_infected) κάθε συνόλου δεδομένων ορίστηκαν σε 0.1 και 0.05. Στα παρακάτω διαγράμματα, για κάθε σύνολο δεδομένων, φαίνεται το πλήθος τον μολυσμένων κόμβων που προέκυψαν για κάθε μία από τις 500 προσομοιώσεις του μοντέλου επιδημίας SI.



Εικόνα 4.1 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων CollegeMsg



Εικόνα 4.2 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων Facebook.



Εικόνα 4.3 Πλήθος των κόμβων που μολύνθηκαν σε κάθε προσομοίωση του μοντέλου SI που πραγματοποιήθηκε για το σύνολο δεδομένων Wiki-Votes.

Μετά την εφαρμογή 500 προσομοιώσεων του επιδημιολογικού μοντέλου SI για κάθε σύνολο δεδομένων, προέκυψαν τα ακόλουθα σύνολα.

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 4.4 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το CollegeMsg. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν

Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα

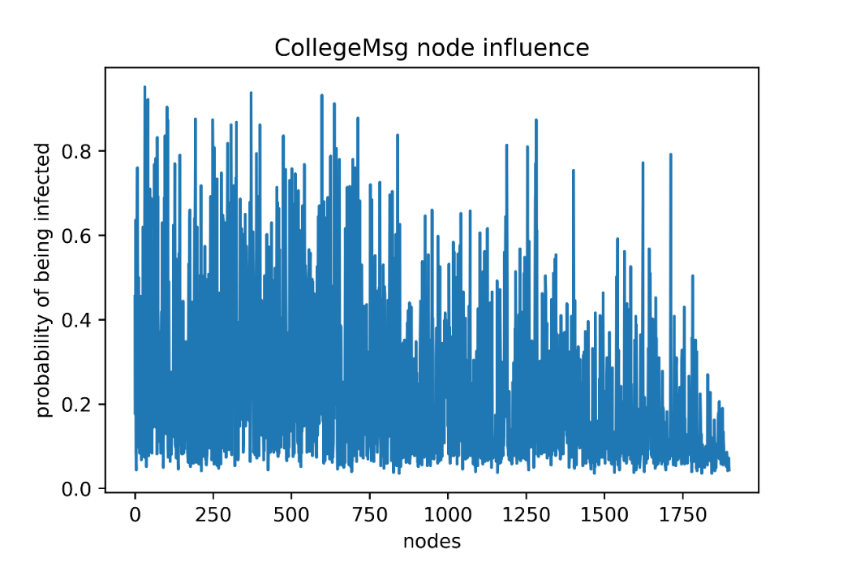
Εικόνα 4.5 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το facebook. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν

Εικόνα που περιέχει κείμενο

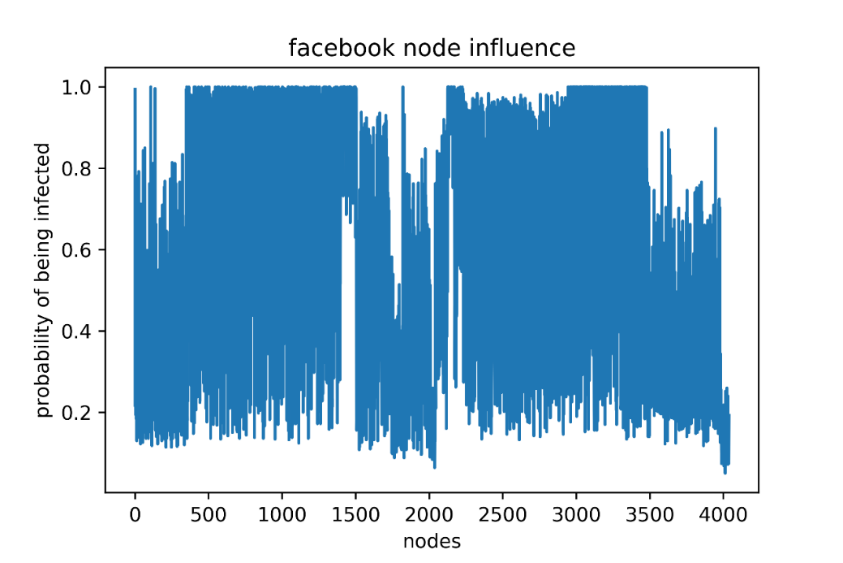
Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 4.6 Σύνολο δεδομένων που προέκυψε μετά τις 10 πρώτες προσομοιώσεις για το Wiki-Votes. Στην πρώτη στήλη φαίνεται το μέγεθος της εξάπλωσης με βάση το μοντέλο SI ενώ στην δεύτερη στήλη φαίνονται αναλυτικά οι κόμβοι του δικτύου που μολύνθηκαν

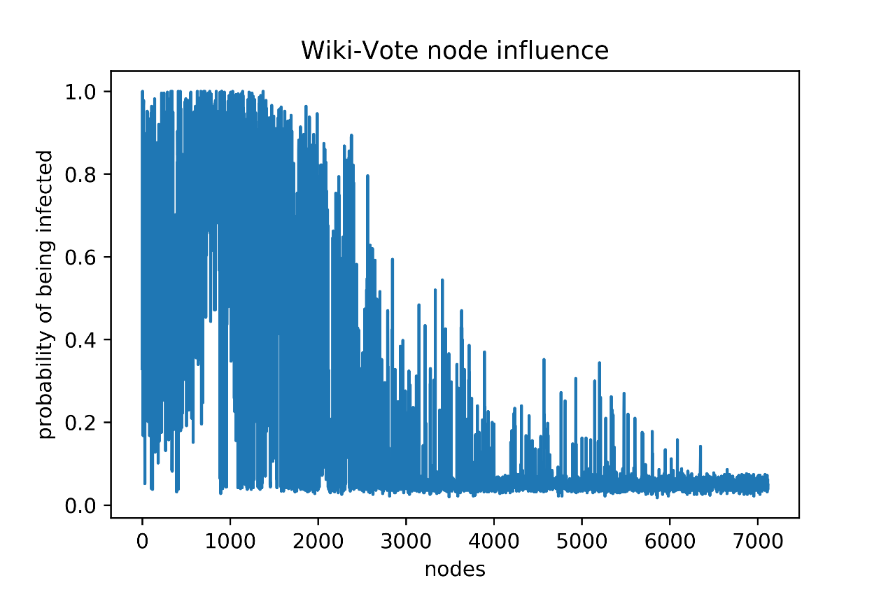
Στην συνέχεια, χρησιμοποιώντας την δεύτερη στήλη του σχήματος, δηλαδή την αναλυτική καταγραφή των κόμβων που μολύνθηκαν σε κάθε προσομοίωση, υπολογίστηκε για κάθε κόμβο η συχνότητα μόλυνσης του. Με βάση την τιμή που προέκυψε για κάθε κόμβο έγινε η τελική ανάθεση της ετικέτας επιρροής. Το πείραμα έγινε για αριθμό ετικετών ίσο με 3 και για αριθμό ετικετών ίσο με 4. Στις παρακάτω εικόνες φαίνεται η συχνότητα εμφάνισης κάθε κόμβου για κάθε ένα από τα σύνολα δεδομένων.



Εικόνα 4.7 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων CollegeMsg



Εικόνα 4.8 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων Facebook



Εικόνα 4.9 Πιθανότητα μόλυνσης κάθε κόμβου για το σύνολο δεδομένων Wiki-Vote

Γνωρίζοντας πλέον την ετικέτα κάθε κόμβου, δηλαδή τις τιμές που θα χρησιμοποιηθούν σαν χαρακτηριστικά στόχοι για τα μοντέλα ταξινόμησης, πρέπει να προσδιοριστούν τα δεδομένα εκπαίδευσης. Για τον προσδιορισμό αυτών, σε αυτή την μέθοδο, χρησιμοποιήθηκε ο αλγόριθμος απεικόνισης node2vec με τη οποία, για κάθε κόμβο, δημιουργήθηκε ένα διάνυσμα αριθμών το οποίο τον αντιπροσωπεύει. Αυτά τα διανύσματα αποτελούν τα δεδομένα εκπαίδευσης που θα χρησιμοποιηθούν στα μοντέλα ταξινόμησης.

Τα δεδομένα εκπαίδευσης που προέκυψαν για κάθε σύνολο δεδομένων, μετά από αφαίρεση μερικών παραδειγμάτων από τις κυρίαρχες κλάσεις με στόχο την βελτίωση της ακρίβειας ταξινόμησης, φαίνονται στα παρακάτω σχήματα, όπου στην τελευταία στήλη κάθε σειράς απεικονίζεται η ετικέτα του κόμβου.

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.10 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Node2vec για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής)

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.11 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Node2vec για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής)

Στις παρακάτω εικόνες φαίνεται η τελική κατανομή των παραδειγμάτων ανά κλάση για κάθε σύνολο δεδομένων.

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.12 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (3 κλάσεις επιρροής)

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.13 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (4 κλάσεις επιρροής)

### 4.1.2 Απεικόνιση κόμβων και αλγόριθμοι ταξινόμησης

Για την ταξινόμηση των κόμβων με κριτήριο την επιρροή τους, δοκιμάστηκαν έξι αλγόριθμοι μηχανικής μάθησης με την βοήθεια της μεθόδου K-Fold Cross Validation οπού το Κ ορίστηκε ως 10. Παράλληλα για την αναλυτικότερη παρουσίαση αποτελεσμάτων για κάθε ετικέτα-κλάση ξεχωριστά, τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου χωρίστηκαν σε 85% και 15% ολόκληρου του συνόλου δεδομένων αντίστοιχα και δημιουργήθηκαν οι πίνακες σύγχυσης και αποτελεσματικότητας για κάθε μέθοδο ταξινόμησης ώστε να φαίνεται η αποτελεσματικότητα στην ταξινόμηση των παραδειγμάτων κάθε κλάσης. Οι έξι αλγόριθμοι μηχανικής μάθησης που χρησιμοποιήθηκαν για κάθε dataset είναι οι εξής:

* GaussianNB
* Decision Tree
* K-nearest Neighbors
* Logistic regression
* Svm
* Neural network

Στις εικόνες που ακολουθούν, για κάθε σύνολο δεδομένων, παρουσιάζονται διαγράμματα στα οποία φαίνεται η σύγκριση των μετρικών αποτελεσματικότητας της ταξινόμησης κάθε αλγορίθμου.

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.14 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (3 κλάσεις επιρροής)

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.15 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Node2vec (4 κλάσεις επιρροής)

Στις εικόνες 14Α και 15Α φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου CollegeMsg σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.68 | 0.663 | 0.641 | 0.641 |
| **Decision Tree** | 0.551 | 0.5 | 0.5 | 0.498 |
| **K-nearest neighbors** | 0.579 | 0.581 | 0.46 | 0.454 |
| **Logistic regression** | 0.734 | 0.702 | 0.691 | 0.694 |
| **Svm** | 0.799 | 0.786 | 0.734 | 0.744 |
| **Neural Network** | 0.783 | 0.758 | 0.739 | 0.744 |

Πίνακας 4.1 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset CollegeMsg (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.16 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το CollegeMsg dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.55 | 0.541 | 0.499 | 0.502 |
| **Decision Tree** | 0.463 | 0.392 | 0.388 | 0.388 |
| **K-nearest neighbors** | 0.497 | 0.484 | 0.382 | 0.376 |
| **Logistic regression** | 0.619 | 0.585 | 0.575 | 0.576 |
| **Svm** | 0.69 | 0.668 | 0.587 | 0.599 |
| **Neural Network** | 0.684 | 0.636 | 0.609 | 0.617 |

Πίνακας 4.2 Αποτελέσματα για συνδυασμό node2vec με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.17 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το CollegeMsg dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) με εξίσου καλή αυτή του Νευρωνικού δικτύου. Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού δικτύου με εξίσου καλή αυτή της μεθόδου Support Vector Machine (Svm). Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Μεταξύ των δύο περιπτώσεων παρατηρούμε μεγάλες διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει απογοητευτικά αποτελέσματα. Αυτό μπορεί να οφείλεται, στην μείωση των παραδειγμάτων κάθε κλάσης εξαιτίας της μίας επιπλέον κλάσης, στην μορφολογία του δικτύου ή στην συσχέτιση που υπάρχει μεταξύ των embeddings του node2vec και των ετικετών.

Στις εικόνες 14Β και 15Β αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου Facebook σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.636 | 0.638 | 0.639 | 0.634 |
| **Decision Tree** | 0.547 | 0.536 | 0.536 | 0.535 |
| **K-nearest neighbors** | 0.747 | 0.762 | 0.723 | 0.727 |
| **Logistic regression** | 0.722 | 0.707 | 0.707 | 0.705 |
| **Svm** | 0.82 | 0.819 | 0.818 | 0.816 |
| **Neural Network** | 0.806 | 0.794 | 0.796 | 0.794 |

Πίνακας 4.3 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Facebook (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.18 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Facebook dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.582 | 0.57 | 0.568 | 0.567 |
| **Decision Tree** | 0.459 | 0.446 | 0.446 | 0.445 |
| **K-nearest neighbors** | 0.661 | 0.678 | 0.642 | 0.64 |
| **Logistic regression** | 0.637 | 0.612 | 0.615 | 0.612 |
| **Svm** | 0.771 | 0.773 | 0.759 | 0.761 |
| **Neural Network** | 0.727 | 0.712 | 0.711 | 0.711 |

Πίνακας 4.4 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Facebook (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.19 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Facebook dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο. Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι GaussianNB και Decision Tree.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm). Η λιγότερο αποτελεσματική μέθοδος για αυτό το dataset αποδείχτηκε ότι είναι η Decision Tree.

Στο συγκεκριμένο σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, παρατηρούμε μικρότερες διαφορές στα αποτελέσματα ταξινόμησης σε σχέση με το dataset CollegeMsg dataset. Η περίπτωση των 4 κλάσεων δίνει μεν μικρότερα αποτελέσματα από ότι αυτή των 3 κλάσεων, αλλά υπάρχει μέθοδος (Svm) που τα αποτελέσματα της είναι ικανοποιητικά. Η μείωση της πτώσης των αποτελεσμάτων, μεταξύ των δύο περιπτώσεων, οφείλεται πιθανότατα στην μεγαλύτερη ισορροπία που υπάρχει στην κατανομή των δεδομένων κάθε κλάσης που φαίνεται στην εικόνα 13Γ.

Στις εικόνες 14Γ και 15Γ αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του τρίτου συνόλου Wiki-Vote σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.76 | 0.721 | 0.701 | 0.702 |
| **Decision Tree** | 0.692 | 0.625 | 0.622 | 0.622 |
| **K-nearest neighbors** | 0.719 | 0.691 | 0.627 | 0.638 |
| **Logistic regression** | 0.779 | 0.727 | 0.714 | 0.717 |
| **Svm** | 0.818 | 0.762 | 0.75 | 0.751 |
| **Neural Network** | 0.824 | 0.777 | 0.776 | 0.776 |

Πίνακας 4.5 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Wiki-Vote (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

Εικόνα 4.20 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Wiki-Vote dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.701 | 0.587 | 0.582 | 0.571 |
| **Decision Tree** | 0.619 | 0.509 | 0.505 | 0.505 |
| **K-nearest neighbors** | 0.655 | 0.566 | 0.503 | 0.512 |
| **Logistic regression** | 0.703 | 0.597 | 0.584 | 0.587 |
| **Svm** | 0.752 | 0.663 | 0.645 | 0.65 |
| **Neural Network** | 0.742 | 0.648 | 0.642 | 0.643 |

Πίνακας 4.6 Αποτελέσματα για συνδυασμό node2vec με μηχανικη μάθηση στο dataset Wiki-Vote (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.21 Πίνακες σύγχησης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και node2vec για το Wiki-Vote dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού Δικτύου ενώ εξίσου καλή ταξινόμηση έγινε με την μέθοδο Support Vector Machine. Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm). Η λιγότερο αποτελεσματική μέθοδος για αυτό το dataset αποδείχτηκε ότι είναι η Decision Tree.

Σε αυτό το σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, ομοίως με το σύνολο CollegeMsg, παρατηρούμε μεγάλες διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει απογοητευτικά αποτελέσματα. Αυτό μπορεί να οφείλεται στην μείωση των παραδειγμάτων κάθε κλάσης εξαιτίας της μίας επιπλέον που προστέθηκε, στην μεγάλη ανισορροπία μεταξύ των κλάσεων, στην μορφολογία του δικτύου ή στην συσχέτιση που υπάρχει μεταξύ των embeddings του node2vec και των ετικετών.

### 4.1.3 Συμπεράσματα συνδυασμού Node2vec και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων

Από τα παραπάνω αποτελέσματα προκύπτουν τα εξής συμπεράσματα:

* Για την πρόβλεψη της επιρροής των κόμβων συνδυάζοντας την μέθοδο Node2vec με αλγόριθμους ταξινόμησης, οι πιο αποτελεσματικοί αλγόριθμοι είναι ο Svm και το Νευρωνικό δίκτυο με τα αποτελέσματα τους να είναι κοντινά.
* Οι λιγότερο αποτελεσματικοί αλγόριθμοι ταξινόμησης αποδείχτηκαν οι k-nearest neighbors και decision tree.
* Η αποτελεσματικότητα του πειράματος για κάθε σύνολο δεδομένων που χρησιμοποιήθηκε δεν εξαρτάται από το πλήθος των κόμβων του δικτύου, αλλά από την μορφολογία αυτού (π.χ. ενώ το δίκτυο Wiki-Vote είναι μεγαλύτερο από το Facebook οι μετρικές αποτελεσματικότητας έχουν μικρότερες τιμές).
* Αν θεωρήσουμε ότι, για το συγκεκριμένο πρόβλημα, οι ικανοποιητικές τιμές για τις μετρικές αποτελεσματικότητας είναι μεγαλύτερες ή περίπου ίσες με 75% τότε η περίπτωση ταξινόμησης σε τέσσερις κλάσεις επιρροής δεν κρίνεται ικανοποιητικά αποτελεσματική (μόνο για το dataset Facebook φτάνει σε τέτοιες τιμές).

## 4.2 Ταξινόμηση κόμβων συνδυάζοντας DeepWalk και μηχανική μάθηση

Σε αυτή την ενότητα προτείνεται η δεύτερη μέθοδος για τον χαρακτηρισμό των κόμβων, με βάση την επιρροή που ασκούν στα κοινωνικά δίκτυα. Ομοίως με την πρώτη, ο χαρακτηρισμός των κόμβων αντιμετωπίζεται ως πρόβλημα ταξινόμησης, συνεπώς για τον χαρακτηρισμό των κόμβων χρησιμοποιούνται αλγόριθμοι ταξινόμησης και πιο συγκεκριμένα αλγόριθμοι επιβλεπόμενης ταξινόμησης. Για την δημιουργία δεδομένων εκπαίδευσης, για τα μοντέλα ταξινόμησης, σε αντίθεση με την πρώτη χρησιμοποιήθηκε η μέθοδος DeepWalk η οποία είναι μια διαφορετική μέθοδος απεικόνισης κόμβων σε χαμηλότερες διαστάσεις.

### 4.2.1 Κατασκευή δεδομένων εκπαίδευσης

Για την κατασκευή των δεδομένων εκπαίδευσης ακολουθήθηκε η ίδια διαδικασία με το πρώτο πείραμα, με την διαφορά ότι χρησιμοποιήθηκε ο αλγόριθμος DeepWalk για την απεικόνιση των κόμβων. Δηλαδή οι ετικέτες επιρροής για κάθε κόμβο είναι αυτές που υπολογίστηκαν στην ενότητα 4.1.1. Τα σύνολα δεδομένων που προέκυψαν φαίνονται στις παρακάτω εικόνες, όπου και εδώ στην τελευταία στήλη κάθε σειράς απεικονίζεται η ετικέτα κάθε κόμβου.

|  |
| --- |
| Εικόνα που περιέχει κείμενο, πληκτρολόγιο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.22 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και DeepWalk για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής)

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.23 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και DeepWalk για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής)

Παράλληλα μετά την αφαίρεση κάποιων παραδειγμάτων από τις κυρίαρχες κλάσεις με στόχο την βελτίωση της ακρίβειας ταξινόμησης, οι τελικές κατανομές παραδειγμάτων ανά κλάση είναι η ακόλουθες.

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.24 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (3 κλάσεις επιρροής)

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.25 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (4 κλάσεις επιρροής)

### 4.2.2 DeepWalk και αλγόριθμοι ταξινόμησης

Για την ταξινόμηση των κόμβων με κριτήριο την επιρροή τους, δοκιμάστηκαν έξι αλγόριθμοι μηχανικής μάθησης με την βοήθεια της μεθόδου K-Fold Cross Validation οπού το Κ ορίστηκε ως 10. Παράλληλα για την αναλυτικότερη παρουσίαση αποτελεσμάτων για κάθε ετικέτα-κλάση ξεχωριστά, τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου χωρίστηκαν σε 85% και 15% ολόκληρου του συνόλου δεδομένων αντίστοιχα και δημιουργήθηκαν οι πίνακες σύγχυσης και αποτελεσματικότητας για κάθε μέθοδο ταξινόμησης ώστε να φαίνεται η αποτελεσματικότητα στην ταξινόμηση των παραδειγμάτων κάθε κλάσης. Οι έξι αλγόριθμοι μηχανικής μάθησης που χρησιμοποιήθηκαν για κάθε dataset είναι οι εξής:

* GaussianNB
* Decision Tree
* K-nearest Neighbors
* Logistic regression
* Svm
* Neural network

Στις εικόνες που ακολουθούν, για κάθε σύνολο δεδομένων, παρουσιάζονται διαγράμματα στα οποία φαίνεται η σύγκριση των μετρικών αποτελεσματικότητας της ταξινόμησης κάθε αλγορίθμου.

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.26 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (3 κλάσεις επιρροής)

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.27 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και DeepWalk (4 κλάσεις επιρροής)

Στις εικόνες 26Α και 27Α φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου CollegeMsg σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.706 | 0.694 | 0.708 | 0.69 |
| **Decision Tree** | 0.566 | 0.51 | 0.503 | 0.504 |
| **K-nearest neighbors** | 0.557 | 0.544 | 0.415 | 0.385 |
| **Logistic regression** | 0.713 | 0.681 | 0.667 | 0.67 |
| **Svm** | 0.814 | 0.798 | 0.767 | 0.776 |
| **Neural Network** | 0.769 | 0.743 | 0.711 | 0.72 |

Πίνακας 4.7 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.28 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το CollegeMsg dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.578 | 0.518 | 0.564 | 0.52 |
| **Decision Tree** | 0.528 | 0.422 | 0.436 | 0.424 |
| **K-nearest neighbors** | 0.592 | 0.482 | 0.548 | 0.488 |
| **Logistic regression** | 0.655 | 0.572 | 0.572 | 0.567 |
| **Svm** | 0.666 | 0.591 | 0.551 | 0.559 |
| **Neural Network** | 0.656 | 0.561 | 0.572 | 0.56 |

Πίνακας 4.8 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.29 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το CollegeMsg dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm). Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο K-nearest neighbors.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού δικτύου με εξίσου καλή αυτή της μεθόδου Support Vector Machine (Svm). Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Και σε αυτό το πείραμα, για το dataset CollegeMsg, μεταξύ των δύο περιπτώσεων παρατηρούμε μεγάλες διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει πολύ χαμηλά αποτελέσματα ταξινόμησης. Αυτό μπορεί να οφείλεται στα αίτια που παρατέθηκαν στο αντίστοιχο τμήμα της ενότητας 4.1.2.

Στις εικόνες 26Β και 27Β αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου Facebook σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.665 | 0.664 | 0.663 | 0.661 |
| **Decision Tree** | 0.535 | 0.526 | 0.527 | 0.526 |
| **K-nearest neighbors** | 0.791 | 0.79 | 0.778 | 0.774 |
| **Logistic regression** | 0.708 | 0.697 | 0.699 | 0.697 |
| **Svm** | 0.835 | 0.839 | 0.836 | 0.834 |
| **Neural Network** | 0.822 | 0.818 | 0.817 | 0.817 |

Πίνακας 4.9 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Facebook (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.30 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Facebook dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.601 | 0.59 | 0.588 | 0.586 |
| **Decision Tree** | 0.445 | 0.431 | 0.432 | 0.43 |
| **K-nearest neighbors** | 0.713 | 0.709 | 0.688 | 0.68 |
| **Logistic regression** | 0.637 | 0.614 | 0.617 | 0.613 |
| **Svm** | 0.765 | 0.762 | 0.754 | 0.754 |
| **Neural Network** | 0.753 | 0.74 | 0.74 | 0.739 |

Πίνακας 4.10 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Facebook (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.31 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Facebook dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο. Η λιγότερο αποτελεσματική μέθοδος για αυτό το dataset αποδείχτηκε ότι είναι η Decision Tree.

Στο συγκεκριμένο σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, παρατηρούμε μικρότερες διαφορές στα αποτελέσματα ταξινόμησης σε σχέση με το dataset CollegeMsg dataset. Η περίπτωση των 4 κλάσεων δίνει μεν μικρότερα αποτελέσματα από ότι αυτή των 3 κλάσεων, αλλά υπάρχουν μέθοδοι (Svm, Νευρωνικό δίκτυο) που τα αποτελέσματα τους είναι ικανοποιητικά. Η μείωση της πτώσης των αποτελεσμάτων, μεταξύ των δύο περιπτώσεων, οφείλεται πιθανότατα στην μεγαλύτερη ισορροπία που υπάρχει στην κατανομή των δεδομένων κάθε κλάσης που φαίνεται στην εικόνα 25Β.

Στις εικόνες 26Γ και 27Γ αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του τρίτου συνόλου Wiki-Vote σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.779 | 0.702 | 0.672 | 0.679 |
| **Decision Tree** | 0.729 | 0.622 | 0.622 | 0.621 |
| **K-nearest neighbors** | 0.764 | 0.697 | 0.618 | 0.64 |
| **Logistic regression** | 0.798 | 0.713 | 0.692 | 0.698 |
| **Svm** | 0.839 | 0.752 | 0.743 | 0.745 |
| **Neural Network** | 0.841 | 0.761 | 0.758 | 0.759 |

Πίνακας 4.11 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.32 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Wiki-Vote dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.784 | 0.549 | 0.545 | 0.531 |
| **Decision Tree** | 0.733 | 0.473 | 0.472 | 0.471 |
| **K-nearest neighbors** | 0.78 | 0.581 | 0.471 | 0.503 |
| **Logistic regression** | 0.803 | 0.592 | 0.549 | 0.563 |
| **Svm** | 0.828 | 0.656 | 0.602 | 0.607 |
| **Neural Network** | 0.839 | 0.648 | 0.64 | 0.642 |

Πίνακας 4.12 Αποτελέσματα για συνδιασμό DeepWalk με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.33 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και DeepWalk για το Wiki-Vote dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού Δικτύου ενώ εξίσου καλή ταξινόμηση έγινε με την μέθοδο Support Vector Machine. Οι λιγότερο αποτελεσματικοί αλγόριθμοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού δικτύου. Οι λιγότερο αποτελεσματικές μέθοδοι για αυτό το dataset αποδείχτηκε ότι είναι οι K-nearest neighbors και Decision Tree.

Σε αυτό το σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, ομοίως με το σύνολο CollegeMsg, παρατηρούμε μεγάλες διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει χαμηλά αποτελέσματα ταξινόμησης. Οι λόγοι αυτής της διαφοράς αναφέρονται στο αντίστοιχο τμήμα της ενότητας 4.1.2.

### 4.2.3 Συμπεράσματα συνδυασμού DeepWalk και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων

Από τα παραπάνω αποτελέσματα προκύπτουν τα εξής συμπεράσματα:

* Για την πρόβλεψη της επιρροής των κόμβων συνδυάζοντας την μέθοδο DeepWalk με αλγόριθμους ταξινόμησης, οι πιο αποτελεσματικοί αλγόριθμοι είναι ο Svm και το Νευρωνικό δίκτυο με τα αποτελέσματα τους να είναι κοντινά.
* Ο λιγότερο αποτελεσματικός αλγόριθμος ταξινόμησης αποδείχτηκε οι k- decision tree.
* Η αποτελεσματικότητα του πειράματος για κάθε σύνολο δεδομένων που χρησιμοποιήθηκε δεν εξαρτάται από το πλήθος των κόμβων του δικτύου, αλλά από την μορφολογία αυτού (π.χ. ενώ το δίκτυο Wiki-Vote είναι μεγαλύτερο από το Facebook οι μετρικές αποτελεσματικότητας έχουν μικρότερες τιμές).
* Αν θεωρήσουμε ότι, για το συγκεκριμένο πρόβλημα, οι ικανοποιητικές τιμές για τις μετρικές αποτελεσματικότητας είναι μεγαλύτερες ή περίπου ίσες με 75% τότε η περίπτωση ταξινόμησης σε τέσσερις κλάσεις επιρροής δεν κρίνεται ικανοποιητικά αποτελεσματική (μόνο για το dataset Facebook φτάνει σε τέτοιες τιμές).

## 4.3 Ταξινόμηση κόμβων συνδυάζοντας Walklets και μηχανική μάθηση

Σε αυτή την ενότητα προτείνεται η τρίτη μέθοδος για τον χαρακτηρισμό των κόμβων, με βάση την επιρροή που ασκούν στα κοινωνικά δίκτυα. Ομοίως με υπόλοιπες μεθόδους που εξετάστηκαν, ο χαρακτηρισμός των κόμβων αντιμετωπίζεται ως πρόβλημα ταξινόμησης, συνεπώς για τον χαρακτηρισμό των κόμβων χρησιμοποιούνται αλγόριθμοι ταξινόμησης και πιο συγκεκριμένα αλγόριθμοι επιβλεπόμενης ταξινόμησης. Σε αυτή την μέθοδο για την δημιουργία δεδομένων εκπαίδευσης, για τα μοντέλα ταξινόμησης χρησιμοποιήθηκε η μέθοδος Walklets η οποία είναι μια παραλλαγή της μεθόδου DeepWalk.

### 4.3.1 Κατασκευή δεδομένων εκπαίδευσης

Για την κατασκευή των δεδομένων εκπαίδευσης ακολουθήθηκε η ίδια διαδικασία με τα προηγούμενα πειράματα, με την διαφορά ότι χρησιμοποιήθηκε ο αλγόριθμος Walklets για την απεικόνιση των κόμβων. Δηλαδή οι ετικέτες επιρροής για κάθε κόμβο είναι αυτές που υπολογίστηκαν στην ενότητα 4.1.1. Τα σύνολα δεδομένων που προέκυψαν φαίνονται στις παρακάτω εικόνες, όπου και εδώ στην τελευταιά στήλη κάθε σειράς απεικονίζεται η ετικέτα κάθε κόμβου.

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.34 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Walklets για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής)

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.35 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και Walklets για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής)

Παράλληλα μετά την αφαίρεση κάποιων παραδειγμάτων από τις κυρίαρχες κλάσεις με στόχο την βελτίωση της ακρίβειας ταξινόμησης, οι τελικές κατανομές παραδειγμάτων ανά κλάση είναι η ακόλουθες.

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.36 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (3 κλάσεις επιρροής)

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.37 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (4 κλάσεις επιρροής)

### 4.3.2 Walklets και αλγόριθμοι ταξινόμησης

Για την ταξινόμηση των κόμβων με κριτήριο την επιρροή τους, δοκιμάστηκαν έξι αλγόριθμοι μηχανικής μάθησης με την βοήθεια της μεθόδου K-Fold Cross Validation οπού το Κ ορίστηκε ως 10. Παράλληλα για την αναλυτικότερη παρουσίαση αποτελεσμάτων για κάθε ετικέτα-κλάση ξεχωριστά, τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου χωρίστηκαν σε 85% και 15% ολόκληρου του συνόλου δεδομένων αντίστοιχα και δημιουργήθηκαν οι πίνακες σύγχυσης και αποτελεσματικότητας για κάθε μέθοδο ταξινόμησης ώστε να φαίνεται η αποτελεσματικότητα στην ταξινόμηση των παραδειγμάτων κάθε κλάσης. Οι έξι αλγόριθμοι μηχανικής μάθησης που χρησιμοποιήθηκαν για κάθε dataset είναι οι εξής:

* GaussianNB
* Decision Tree
* K-nearest Neighbors
* Logistic regression
* Svm
* Neural network

Στις εικόνες που ακολουθούν, για κάθε σύνολο δεδομένων, παρουσιάζονται διαγράμματα στα οποία φαίνεται η σύγκριση των μετρικών αποτελεσματικότητας της ταξινόμησης κάθε αλγορίθμου.

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.38 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (3 κλάσεις επιρροής)

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.39 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και Walklets (4 κλάσεις επιρροής)

Στις εικόνες 38A και 39A φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου CollegeMsg σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.445 | 0.483 | 0.524 | 0.417 |
| **Decision Tree** | 0.699 | 0.662 | 0.66 | 0.657 |
| **K-nearest neighbors** | 0.757 | 0.747 | 0.676 | 0.685 |
| **Logistic regression** | 0.777 | 0.753 | 0.747 | 0.746 |
| **Svm** | 0.78 | 0.767 | 0.735 | 0.741 |
| **Neural Network** | 0.818 | 0.796 | 0.796 | 0.793 |

Πίνακας 4.13 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.40 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το CollegeMsg dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.436 | 0.449 | 0.503 | 0.424 |
| **Decision Tree** | 0.562 | 0.485 | 0.475 | 0.476 |
| **K-nearest neighbors** | 0.542 | 0.426 | 0.381 | 0.357 |
| **Logistic regression** | 0.695 | 0.645 | 0.621 | 0.626 |
| **Svm** | 0.753 | 0.748 | 0.677 | 0.698 |
| **Neural Network** | 0.752 | 0.733 | 0.682 | 0.694 |

Πίνακας 4.14 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.41 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το CollegeMsg dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού Δικτύου. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο GaussianNB.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η, οριακά, καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm). Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο K-nearest neighbors.

Και σε αυτό το πείραμα, για το dataset, μεταξύ των δύο περιπτώσεων παρατηρούμε σημαντικές διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει εξαιρετικά χαμηλά αποτελέσματα ταξινόμησης. Αυτό μπορεί να οφείλεται, στην μορφολογία του δικτύου, στην μείωση των παραδειγμάτων κάθε κλάσης εξαιτίας της μίας επιπλέον κλάσης ή στον τρόπο με τον οποίο τοποθετήθηκαν οι ταμπέλες επιρροής στους κόμβους.

Στις εικόνες 38B και 39B αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου Facebook σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.773 | 0.787 | 0.775 | 0.777 |
| **Decision Tree** | 0.77 | 0.769 | 0.771 | 0.769 |
| **K-nearest neighbors** | 0.831 | 0.827 | 0.834 | 0.827 |
| **Logistic regression** | 0.825 | 0.827 | 0.825 | 0.825 |
| **Svm** | 0.882 | 0.888 | 0.883 | 0.883 |
| **Neural Network** | 0.876 | 0.879 | 0.875 | 0.875 |

Πίνακας 4.15 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Facebook (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.42 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Facebook dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.685 | 0.684 | 0.671 | 0.675 |
| **Decision Tree** | 0.706 | 0.692 | 0.692 | 0.691 |
| **K-nearest neighbors** | 0.778 | 0.768 | 0.76 | 0.758 |
| **Logistic regression** | 0.782 | 0.77 | 0.769 | 0.768 |
| **Svm** | 0.842 | 0.833 | 0.833 | 0.833 |
| **Neural Network** | 0.836 | 0.83 | 0.826 | 0.826 |

Πίνακας 4.16 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Facebook (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.43 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Facebook dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree. Σε αντίθεση με τις άλλες μεθόδους απεικόνισης κόμβων που χρησιμοποιήθηκαν, στην συγκεκριμένη προέκυψαν σχετικά καλές ταξινομήσεις για κάθε μέθοδο ταξινόμησης που εξετάστηκε.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο. Η λιγότερο αποτελεσματική μέθοδος για αυτό το dataset αποδείχτηκε ότι είναι η Gaussian NB.

Στο συγκεκριμένο σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, παρατηρούμε μικρότερες διαφορές στα αποτελέσματα ταξινόμησης σε σχέση με το dataset CollegeMsg dataset. Η περίπτωση των 4 κλάσεων δίνει μεν αναμενόμενα μικρότερα αποτελέσματα από ότι αυτή των 3 κλάσεων, αλλά υπάρχουν μέθοδοι (Svm, Νευρωνικό δίκτυο) που τα αποτελέσματα τους είναι αρκετά καλά.

Στις εικόνες 38Γ και 39Γ αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του τρίτου συνόλου Wiki-Vote σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.699 | 0.667 | 0.545 | 0.55 |
| **Decision Tree** | 0.817 | 0.721 | 0.725 | 0.722 |
| **K-nearest neighbors** | 0.836 | 0.761 | 0.719 | 0.731 |
| **Logistic regression** | 0.866 | 0.79 | 0.779 | 0.783 |
| **Svm** | 0.875 | 0.811 | 0.812 | 0.808 |
| **Neural Network** | 0.865 | 0.789 | 0.79 | 0.788 |

Πίνακας 4.17 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.44 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Wiki-Vote dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.64 | 0.455 | 0.481 | 0.451 |
| **Decision Tree** | 0.7 | 0.581 | 0.58 | 0.58 |
| **K-nearest neighbors** | 0.728 | 0.638 | 0.592 | 0.599 |
| **Logistic regression** | 0.771 | 0.67 | 0.669 | 0.667 |
| **Svm** | 0.781 | 0.694 | 0.687 | 0.682 |
| **Neural Network** | 0.766 | 0.672 | 0.666 | 0.666 |

Πίνακας 4.18 Αποτελέσματα για συνδιασμό Walklets με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.45 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και Walklets για το Wiki-Vote dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την μέθοδο Support Vector Machine, ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό Δίκτυο. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Gaussian NB. Όπως και στο dataset Facebook, σε αντίθεση με τις άλλες μεθόδους απεικόνισης κόμβων που χρησιμοποιήθηκαν, στην συγκεκριμένη προέκυψαν σχετικά καλές ταξινομήσεις για κάθε μέθοδο ταξινόμησης που εξετάστηκε με εξαίρεση την GaussianNB.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την μέθοδο Support Vector Machine. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Gaussian NB.

Σε αυτό το σύνολο δεδομένων, μεταξύ των δύο περιπτώσεων, ομοίως με το σύνολο CollegeMsg, παρατηρούμε μεγάλες διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει χαμηλά αποτελέσματα ταξινόμησης. Οι λόγοι αυτής της διαφοράς αναφέρονται στο αντίστοιχο κομμάτι της ενότητας 4.1.2

### 4.3.3 Συμπεράσματα συνδυασμού Walklets και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων

Από τα παραπάνω αποτελέσματα προκύπτουν τα εξής συμπεράσματα:

* Για την πρόβλεψη της επιρροής των κόμβων συνδυάζοντας την μέθοδο Walklets με αλγόριθμους ταξινόμησης, οι πιο αποτελεσματικοί αλγόριθμοι είναι ο Svm και το Νευρωνικό δίκτυο με τα αποτελέσματα τους να είναι κοντινά.
* Ο λιγότερο αποτελεσματικός αλγόριθμος ταξινόμησης αποδείχτηκε ο GaussianNB.
* Η αποτελεσματικότητα του πειράματος για κάθε σύνολο δεδομένων που χρησιμοποιήθηκε δεν εξαρτάται από το πλήθος των κόμβων του δικτύου, αλλά από την μορφολογία αυτού (π.χ. ενώ το δίκτυο Wiki-Vote είναι μεγαλύτερο από το Facebook οι μετρικές αποτελεσματικότητας έχουν μικρότερες τιμές).
* Αν θεωρήσουμε ότι, για το συγκεκριμένο πρόβλημα, οι ικανοποιητικές τιμές για τις μετρικές αποτελεσματικότητας είναι μεγαλύτερες ή περίπου ίσες με 75% τότε η περίπτωση ταξινόμησης σε τέσσερις κλάσεις επιρροής δεν κρίνεται ικανοποιητικά αποτελεσματική (μόνο για το dataset Facebook γίνεται πολύ καλή ταξινόμηση με τιμές μεγαλύτερες του 80%).

## 4.4 Ταξινόμηση κόμβων συνδυάζοντας την επέκταση της μεθόδου Graph2vec για κόμβους (ext-Graph2vec) και τη μηχανική μάθηση

Σε αυτή την ενότητα προτείνεται η τέταρτη και τελευταία μέθοδος που εξετάστηκε για τον χαρακτηρισμό των κόμβων, με βάση την επιρροή που ασκούν στα κοινωνικά δίκτυα. Ομοίως με την υπόλοιπες μεθόδους, ο χαρακτηρισμός των κόμβων αντιμετωπίζεται ως πρόβλημα ταξινόμησης, συνεπώς για τον χαρακτηρισμό των κόμβων χρησιμοποιούνται αλγόριθμοι ταξινόμησης και πιο συγκεκριμένα αλγόριθμοι επιβλεπόμενης ταξινόμησης. Στην παρούσα μέθοδο για την δημιουργία δεδομένων εκπαίδευσης, για τα μοντέλα ταξινόμησης, χρησιμοποιήθηκε μια επέκταση της μεθόδου Graph2vec για κόμβους. Η Graph2vec είναι μια μέθοδος απεικόνισης ολόκληρων γράφων σε χώρους λιγότερων διαστάσεων.

### 4.4.1 Κατασκευή δεδομένων εκπαίδευσης

Για την κατασκευή των δεδομένων εκπαίδευσης ακολουθήθηκε η ίδια διαδικασία με τα προηγούμενα πειράματα, με την διαφορά ότι χρησιμοποιήθηκε μια επέκταση της μεθόδου Graph2vec για την απεικόνιση των κόμβων. Δηλαδή οι ετικέτες επιρροής για κάθε κόμβο είναι αυτές που υπολογίστηκαν στην ενότητα 4.1.1. Τα σύνολα δεδομένων που προέκυψαν φαίνονται στις παρακάτω εικόνες, όπου και εδώ στην τελευταία στήλη κάθε σειράς απεικονίζεται η ετικέτα κάθε κόμβου.

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.46 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(3 κλάσεις επιρροής)

|  |
| --- |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |
| Εικόνα που περιέχει κείμενο  Περιγραφή που δημιουργήθηκε αυτόματα |

Εικόνα 4.47 Τελικά σύνολα εκπαίδευσης, για τον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για τα CollegeMsg, Facebook, Wiki-Vote αντιστοιχα(4 κλάσεις επιρροής)

Παράλληλα μετά την αφαίρεση κάποιων παραδειγμάτων από τις κυρίαρχες κλάσεις με στόχο την βελτίωση της ακρίβειας ταξινόμησης, οι τελικές κατανομές παραδειγμάτων ανά κλάση είναι η ακόλουθες.

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.48 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (3 κλάσεις επιρροής)

|  |  |
| --- | --- |
|  |  |
|  | |

Εικόνα 4.49 Τελική κατανομή παραδειγμάτων ανά κλάση για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (4 κλάσεις επιρροής)

### 4.3.2 Επέκταση μεθόδου Graph2vec και αλγόριθμοι ταξινόμησης

Για την ταξινόμηση των κόμβων με κριτήριο την επιρροή τους, δοκιμάστηκαν έξι αλγόριθμοι μηχανικής μάθησης με την βοήθεια της μεθόδου K-Fold Cross Validation οπού το Κ ορίστηκε ως 10. Παράλληλα για την αναλυτικότερη παρουσίαση αποτελεσμάτων για κάθε ετικέτα-κλάση ξεχωριστά, τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου χωρίστηκαν σε 85% και 15% ολόκληρου του συνόλου δεδομένων αντίστοιχα και δημιουργήθηκαν οι πίνακες σύγχυσης και αποτελεσματικότητας για κάθε μέθοδο ταξινόμησης ώστε να φαίνεται η αποτελεσματικότητα στην ταξινόμηση των παραδειγμάτων κάθε κλάσης. Οι έξι αλγόριθμοι μηχανικής μάθησης που χρησιμοποιήθηκαν για κάθε dataset είναι οι εξής:

* GaussianNB
* Decision Tree
* K-nearest Neighbors
* Logistic regression
* Svm
* Neural network

Στις εικόνες που ακολουθούν, για κάθε σύνολο δεδομένων, παρουσιάζονται διαγράμματα στα οποία φαίνεται η σύγκριση των μετρικών αποτελεσματικότητας της ταξινόμησης κάθε αλγορίθμου.

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.50 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (3 κλάσεις επιρροής)

|  |
| --- |
|  |
|  |
|  |

Εικόνα 4.51 Μετρικές αποτελεσματικότητας ταξινόμησης για κάθε dataset στον συνδυασμό μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους (4 κλάσεις επιρροής)

Στις εικόνες 50Α και 51Α φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου CollegeMsg σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.878 | 0.902 | 0.837 | 0.849 |
| **Decision Tree** | 0.832 | 0.817 | 0.813 | 0.811 |
| **K-nearest neighbors** | 0.892 | 0.892 | 0.868 | 0.874 |
| **Logistic regression** | 0.88 | 0.876 | 0.837 | 0.848 |
| **Svm** | 0.892 | 0.891 | 0.86 | 0.869 |
| **Neural Network** | 0.887 | 0.871 | 0.862 | 0.864 |

Πίνακας 4.19 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset CollegeMsg (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.52 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το CollegeMsg dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.838 | 0.834 | 0.805 | 0.811 |
| **Decision Tree** | 0.781 | 0.77 | 0.761 | 0.762 |
| **K-nearest neighbors** | 0.861 | 0.859 | 0.822 | 0.833 |
| **Logistic regression** | 0.828 | 0.828 | 0.753 | 0.767 |
| **Svm** | 0.864 | 0.859 | 0.838 | 0.844 |
| **Neural Network** | 0.847 | 0.83 | 0.815 | 0.819 |

Πίνακας 4.20 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset CollegeMsg (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.53 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το CollegeMsg dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε, οριακά, με την βοήθεια της μεθόδου K-nearest neighbors με την μέθοδο Svm να δείχνει εξίσου αποτελεσματική. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε το Decision Tree. Χρησιμοποιώντας την συγκεκριμένη μέθοδο απεικόνισης κόμβων, τα αποτελέσματα που προέκυψαν από κάθε ταξινομητή είναι πάρα πολύ καλά.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν η, οριακά, καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) με τα αποτελέσματα της μεθόδου k-nearest neighbors να είναι εξίσου καλά. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree.

Στο συγκεκριμένο πείραμα, για το dataset CollegeMsg, μεταξύ των δύο περιπτώσεων παρατηρούμε διαφορές στα αποτελέσματα ταξινόμησης με την περίπτωση των 4 κλάσεων να δίνει πιο χαμηλά αποτελέσματα ταξινόμησης. Σε αντίθεση όμως με τα άλλα πειράματα, τα αποτελέσματα ταξινόμησης για την περίπτωση των 4 κλάσεων είναι πάρα πολύ καλά.

Στις εικόνες 50Β και 51Β αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του συνόλου Facebook σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.891 | 0.887 | 0.887 | 0.887 |
| **Decision Tree** | 0.837 | 0.832 | 0.832 | 0.831 |
| **K-nearest neighbors** | 0.868 | 0.866 | 0.865 | 0.865 |
| **Logistic regression** | 0.889 | 0.886 | 0.885 | 0.885 |
| **Svm** | 0.894 | 0.89 | 0.89 | 0.89 |
| **Neural Network** | 0.892 | 0.889 | 0.89 | 0.889 |

Πίνακας 4.21 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Facebook (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.54 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Facebook dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.831 | 0.823 | 0.829 | 0.823 |
| **Decision Tree** | 0.772 | 0.763 | 0.764 | 0.762 |
| **K-nearest neighbors** | 0.808 | 0.803 | 0.806 | 0.803 |
| **Logistic regression** | 0.845 | 0.837 | 0.838 | 0.836 |
| **Svm** | 0.842 | 0.834 | 0.836 | 0.833 |
| **Neural Network** | 0.844 | 0.841 | 0.836 | 0.836 |

Πίνακας 4.22 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Facebook (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.55 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Facebook dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια της μεθόδου Support Vector Machine (Svm) ενώ εξίσου καλή ταξινόμηση έγινε με το Νευρωνικό δίκτυο και με την μέθοδο Logistic Regression. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree. Σε αντίθεση με τις άλλες μεθόδους απεικόνισης κόμβων που χρησιμοποιήθηκαν τα αποτελέσματα που προέκυψαν από κάθε ταξινομητή είναι πάρα πολύ καλά.

Στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού δικτυού ενώ εξίσου καλή ταξινόμηση έγινε με την μέθοδο Svm και με την μέθοδο Logistic Regression. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree.

Στο συγκεκριμένο πείραμα για το Facebook, όπως έγινε και με το CollegeMsg dataset, τα αποτελέσματα ταξινόμησης για την περίπτωση των 4 κλάσεων είναι πάρα πολύ καλά, σε αντίθεση με τα υπόλοιπα πειράματα απεικόνισης κόμβων που χρησιμοποιήθηκαν.

Στις εικόνες 50Γ και 51Γ αντίστοιχα φαίνονται τα ποσοστά αποτελεσματικότητας ταξινόμησης (accuracy, precision, recall, f1 score) που πέτυχαν οι αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των κόμβων του τρίτου συνόλου Wiki-Vote σε 3 και 4 κλάσεις αντίστοιχα. Τα αναλυτικά αποτελέσματα καθώς και τα αποτελέσματα ανά κλάση παρουσιάζονται στους πίνακες που ακολουθούν.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.927 | 0.885 | 0.927 | 0.902 |
| **Decision Tree** | 0.936 | 0.903 | 0.902 | 0.902 |
| **K-nearest neighbors** | 0.948 | 0.924 | 0.922 | 0.923 |
| **Logistic regression** | 0.938 | 0.901 | 0.937 | 0.915 |
| **Svm** | 0.939 | 0.903 | 0.935 | 0.916 |
| **Neural Network** | 0.963 | 0.94 | 0.946 | 0.943 |

Πίνακας 4.23 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Wiki-Vote (3 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.56 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Wiki-Vote dataset (για 3 κλάσεις)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| **Gaussian NB** | 0.906 | 0.841 | 0.874 | 0.849 |
| **Decision Tree** | 0.914 | 0.844 | 0.84 | 0.842 |
| **K-nearest neighbors** | 0.929 | 0.873 | 0.867 | 0.868 |
| **Logistic regression** | 0.926 | 0.868 | 0.898 | 0.879 |
| **Svm** | 0.919 | 0.858 | 0.895 | 0.872 |
| **Neural Network** | 0.943 | 0.897 | 0.897 | 0.895 |

Πίνακας 4.24 Αποτελέσματα για συνδιασμό της επέκτασης του Graph2vec για κόμβους με μηχανική μάθηση στο dataset Wiki-Vote (4 κλάσεις)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Εικόνα 4.57 Πίνακες σύγχυσης και πίνακες μετρικών ανά κλάση για πείραμα συνδυασμού μηχανικής μάθησης και επέκτασης μεθόδου Graph2vec για κόμβους για το Wiki-Vote dataset (για 4 κλάσεις)

Από τους παραπάνω πίνακες προκύπτουν τα εξής αποτελέσματα:

Στην περίπτωση με τις 3 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού Δικτύου, ενώ εξίσου καλή ταξινόμηση έγινε με τις μεθόδους Svm και Logistic Regression. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree. Όπως και στο άλλα dataset, σε αντίθεση με τις άλλες μεθόδους απεικόνισης κόμβων που χρησιμοποιήθηκαν, στην συγκεκριμένη προέκυψαν σχετικά καλές ταξινομήσεις για κάθε μέθοδο ταξινόμησης που εξετάστηκε.

Και στην περίπτωση με τις 4 κλάσεις, μεταξύ των 6 μεθόδων ταξινόμησης που χρησιμοποιήθηκαν, οριακά, η καλύτερη ταξινόμηση έγινε με την βοήθεια του Νευρωνικού Δικτύου, ενώ εξίσου καλή ταξινόμηση έγινε με τις μεθόδους Svm και Logistic Regression.. Ο λιγότερο αποτελεσματικός αλγόριθμος για αυτό το dataset αποδείχτηκε ότι είναι ο Decision Tree.

Στο συγκεκριμένο πείραμα και για το Wiki-Vote, όπως έγινε και με τα υπόλοιπα datasets, τα αποτελέσματα ταξινόμησης για την περίπτωση των 4 κλάσεων είναι πάρα πολύ καλά, σε αντίθεση με τα υπόλοιπα πειράματα απεικόνισης κόμβων που χρησιμοποιήθηκαν.

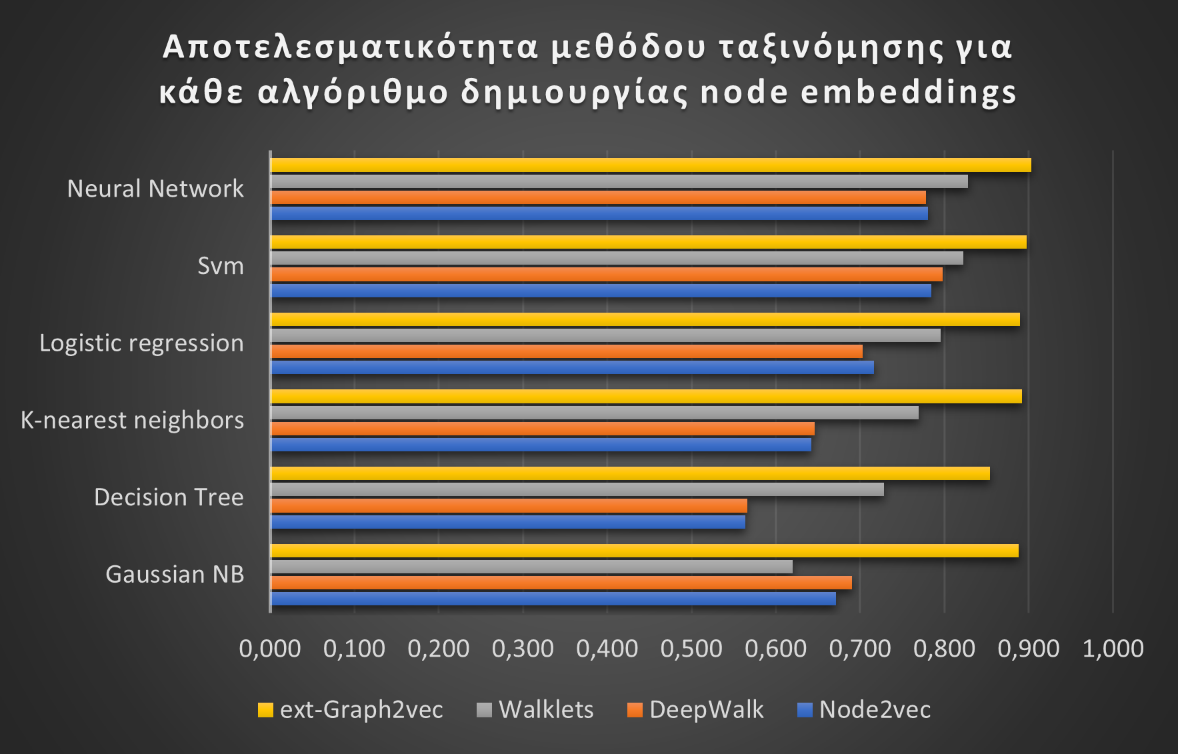
### 4.4.3 Συμπεράσματα συνδυασμού της επέκτασης του Graph2vec για κόμβους και αλγορίθμων ταξινόμησης για την πρόβλεψη της επιρροής των κόμβων

Από τα παραπάνω αποτελέσματα προκύπτουν τα εξής συμπεράσματα:

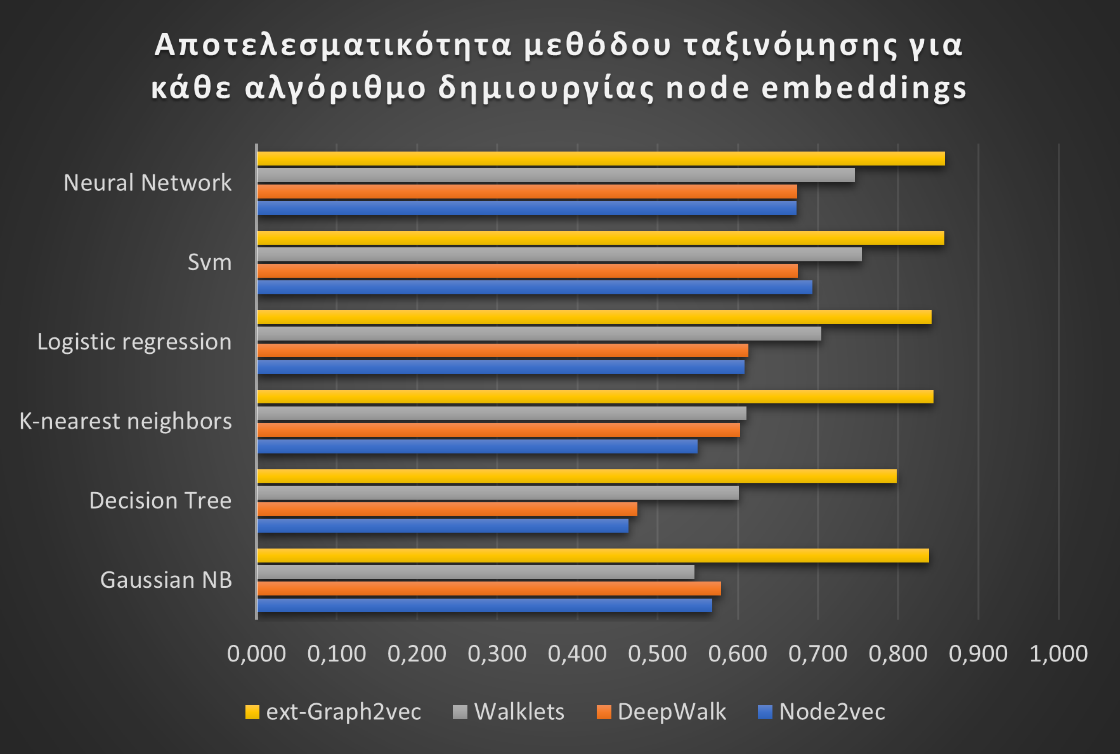
* Για την πρόβλεψη της επιρροής των κόμβων συνδυάζοντας την επέκταση της μεθόδου Graph2vec με αλγόριθμους ταξινόμησης, οι πιο αποτελεσματικοί αλγόριθμοι είναι ο Svm και το Νευρωνικό δίκτυο με τα αποτελέσματα τους να είναι κοντινά.
* Όλοι οι αλγόριθμοι ταξινόμησης που χρησιμοποιήθηκαν έδωσαν πολύ καλά αποτελέσματα και για την περίπτωση με τις τρεις κλάσεις επιρροής και με την περίπτωση με τις τέσσερις κλάσεις επιρροής.
* Η αποτελεσματικότητα του πειράματος για κάθε σύνολο δεδομένων που χρησιμοποιήθηκε φαίνεται να εξαρτάται από το πλήθος των κόμβων του δικτύου αφού το δίκτυο Wiki-Vote που έχει τους περισσότερους κόμβους έχει τα καλύτερά αποτελέσματα.
* Αν θεωρήσουμε ότι, για το συγκεκριμένο πρόβλημα, οι ικανοποιητικές τιμές για τις μετρικές αποτελεσματικότητας είναι μεγαλύτερες ή περίπου ίσες με 75% τότε η περίπτωση ταξινόμησης σε τέσσερις κλάσεις κρίνεται ικανοποιητικά αποτελεσματική αφού για κάθε σύνολο δεδομένων οι τιμές είναι μεγαλύτερες του 80%.

## 4.5 Σύγκριση αποτελεσμάτων

Από τα παραπάνω αποτελέσματα, με την βοήθεια της μέσης αποτελεσματικότητας, προκύπτουν τα παρακάτω διαγράμματα που δείχνουν την αποτελεσματικότητα για κάθε συνδυασμό μεθόδου δημιουργίας node embeddings και μεθόδου ταξινόμησης.

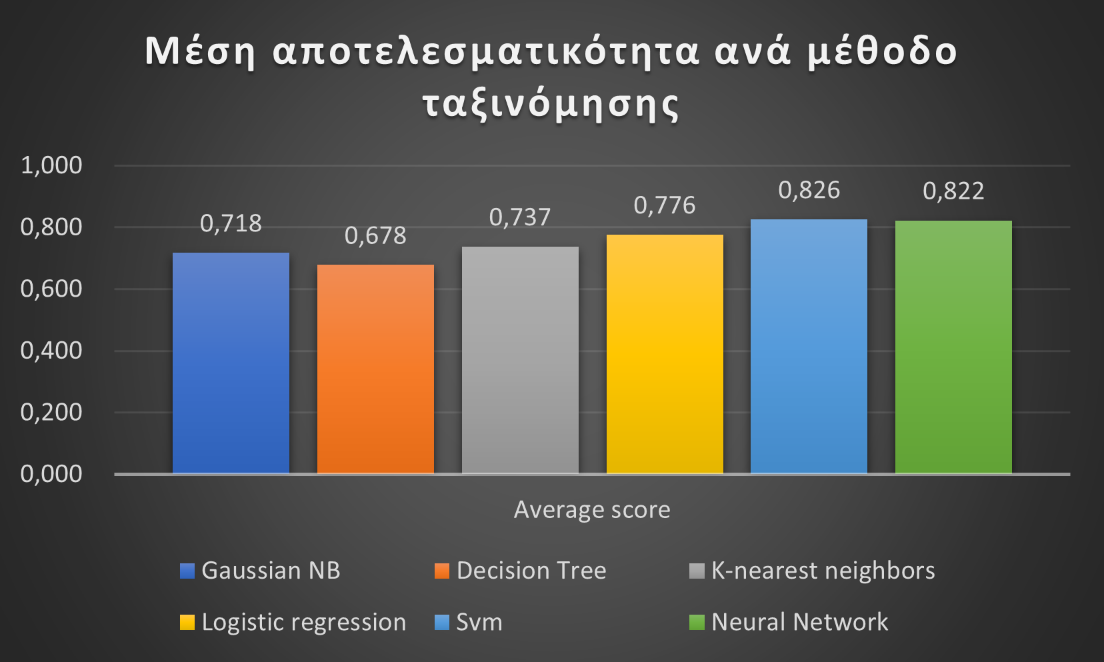


Εικόνα 4.58 Αποτελεσματικότητα μεθόδου ταξινόμησης για κάθε αλγόριθμο δημιουργίας node embeddings για την περίπτωση των 3 κλάσεων

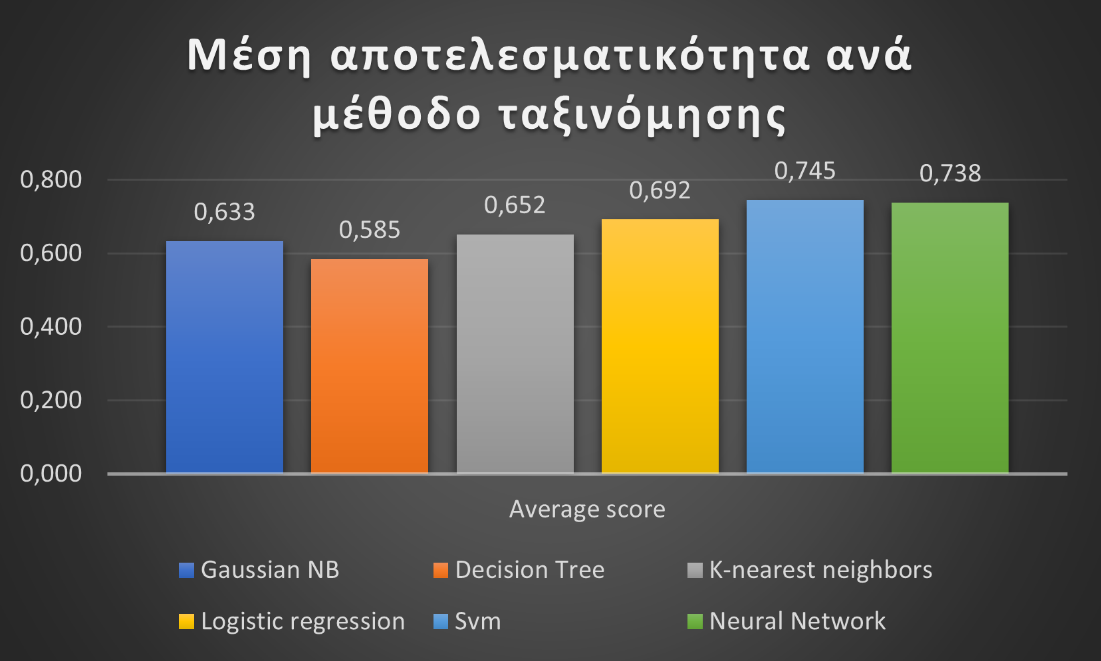


Εικόνα 4.59 Αποτελεσματικότητα μεθόδου ταξινόμησης για κάθε αλγόριθμο δημιουργίας node embeddings για την περίπτωση των 7 κλάσεων

### 4.5.1 Σύγκριση αλγορίθμων ταξινόμησης για το πρόβλεψη της επιρροής σε κοινωνικά δίκτυα



Εικόνα 4.60 Διάγραμμα μέσης αποτελεσματικότητα ανά μέθοδο ταξινόμησης για 3 κλάσεις

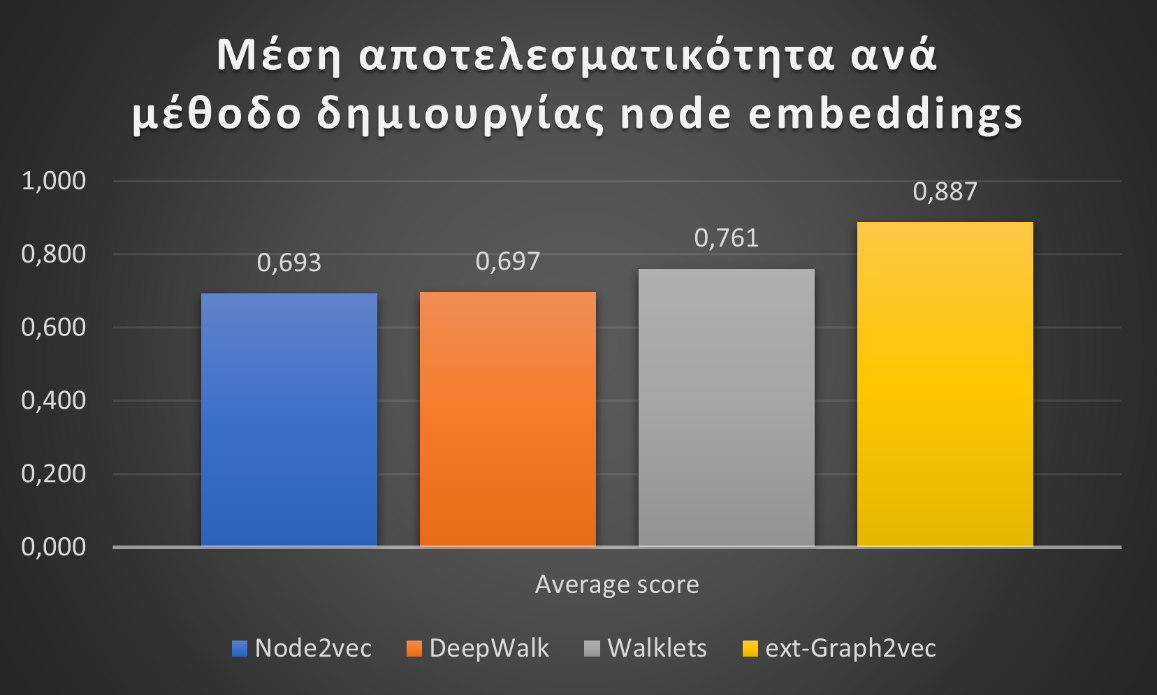


Εικόνα 4.61 Διάγραμμα μέσης αποτελεσματικότητα ανά μέθοδο ταξινόμησης για 3 κλάσεις

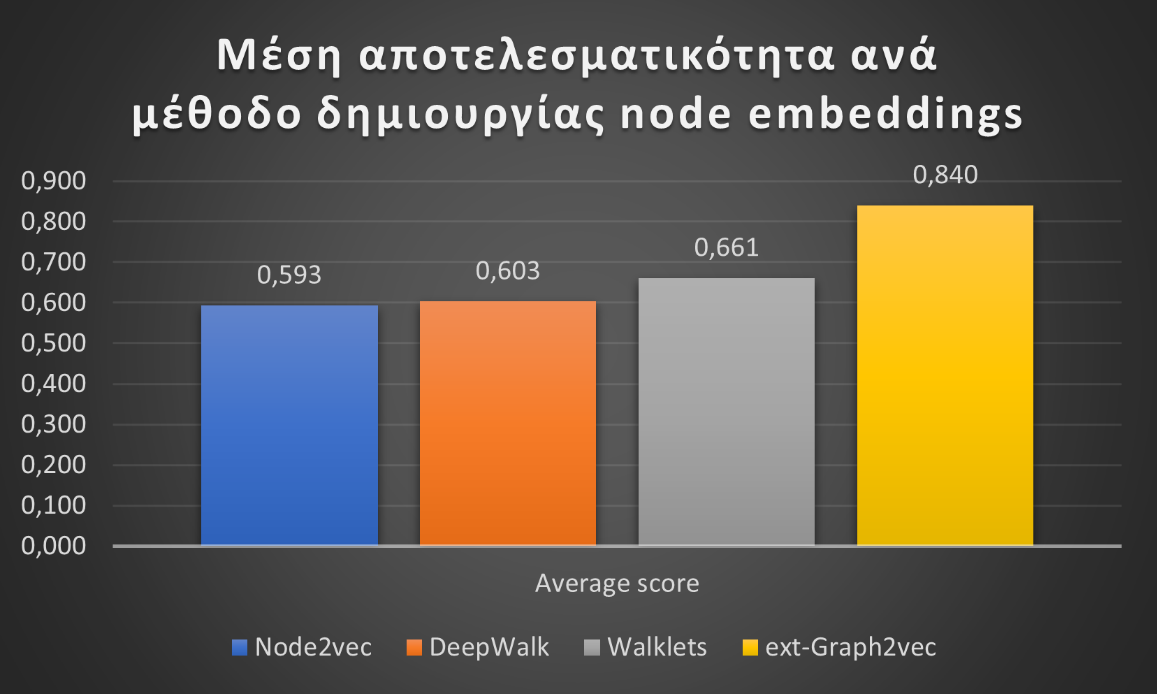
Για την σύγκριση των αλγορίθμων ταξινόμησης στο πρόβλημα της πρόβλεψης της επιρροής των κόμβων σε ένα κοινωνικό δίκτυο με τον τρόπο που αναλύθηκε στην παρούσα διπλωματική εργασία προκύπτουν τα εξής συμπεράσματα:

* Οι πιο αποτελεσματικοί αλγόριθμοι ταξινόμησης είναι ο αλγόριθμος Svm και το Νευρωνικο δίκτυο.
* Ο λιγότερο αποτελεσματικός αλγόριθμος ταξινόμησης είναι τα δέντρα απόφασης (decision trees)
* Με την αύξηση των κλάσεων ταξινόμησης παρατηρείται μεγάλη μείωση της αποτελεσματικότητας (της τάξεως του 10%)
* Οι μοναδικές μέθοδοι που ανταποκρίνονται ικανοποιητικά σε αυτή την αύξηση είναι ο Svm και το Νευρωνικο δίκτυο.

### 4.5.2 Σύγκριση αλγορίθμων δημιουργίας node embeddings για την πρόβλεψη της επιρροής σε κοινωνικά δίκτυα

**

Εικόνα 4.62 Μέση αποτελεσματικότητα ταξινόμησης για κάθε μέθοδο δημιουργίας node embeddings που χρησιμοποιήθηκε για 3 κλάσεις επιρροής

**

Εικόνα 4.63 Μέση αποτελεσματικότητα ταξινόμησης για κάθε μέθοδο δημιουργίας node embeddings που χρησιμοποιήθηκε για 4 κλάσεις επιρροής

Για την σύγκριση των αλγορίθμων δημιουργίας node embeddings στο πρόβλημα της πρόβλεψης της επιρροής των κόμβων σε ένα κοινωνικό δίκτυο με τον τρόπο που αναλύθηκε στην παρούσα διπλωματική εργασία προκύπτουν τα εξής συμπεράσματα:

* Ο πιο αποτελεσματικός, με διαφορά, αλγόριθμος δημιουργίας node embeddings είναι η επέκταση της μεθόδου Graph2vec (ext-Graph2vec).
* Οι λιγότερο αποτελεσματικοί αλγόριθμοι δημιουργίας node embeddings είναι ο node2vec και ο DeepWalk.
* Με την αύξηση των κλάσεων ταξινόμησης παρατηρείται μεγάλη μείωση της αποτελεσματικότητας (της τάξεως του 10%) για τις τρεις από τις τέσσερις μεθόδους. Η μέθοδος που δεν έχει μεγάλη μείωση στην αποτελεσματικότητα (περίπου 5%) είναι η ext-Graph2vec η οποία φαίνεται να κυριαρχεί σε σημείο όπου η μέση αποτελεσματικότητα της για ταξινόμηση σε τέσσερις κλάσεις επιρροής είναι πολύ καλύτερη από την αποτελεσματικότητα των υπολοίπων μεθόδων για τρεις κλάσεις επιρροής.
* Τέλος, στις εικόνες 226 και 227 φαίνεται η υψηλή αποτελεσματικότητα της για όλους τους αλγόριθμους ταξινόμησης που χρησιμοποιήθηκαν (υψηλότερη ή περίπου ίση με 80%)

# Κεφάλαιο 5: Συμπεράσματα

Η κοινωνική επιρροή είναι ένα φαινόμενο που στις μέρες μας συναντάται πέρα από τον πραγματικό κόσμο και στον εικονικό μέσω των κοινωνικών δικτύων. Ουσιαστικά είναι η τάση των ανθρώπων ή στην περίπτωση των κοινωνικών δικτύων των χρηστών να αλλάζουν την συμπεριφορά τους και τις πεποιθήσεις τους με απώτερο στόχο να προσαρμόζονται στις ανάγκες του κοινωνικού συνόλου που ανήκουν. Οι εφαρμογές που έχει εντοπίζονται σε διάφορους τομείς όπως η διαφήμιση, η ενημέρωση, οι εκλογές κ.α. Η πρόβλεψη της κοινωνικής επιρροής, αποτελεί ένα επιστημονικό πεδίο το οποίο ερευνάται όλο και περισσότερο με την πάροδο του χρόνου γιατί έχουν γίνει αντιληπτά τα οφέλη που μπορούν να προκύψουν από αυτή.

## 5.1. Ανασκόπηση εργασίας

Το πρόβλημα της πρόβλεψης της επιρροής, με embeddings, σε κοινωνικά δίκτυα, εξετάστηκε ως ένα πρόβλημα ταξινόμησης. Για να βρεθεί η πραγματική επιρροή των κόμβων, για την δημιουργία των ετικετών επιρροής, χρησιμοποιήθηκε το μοντέλο επιδημίας SI από το όποιο υπολογίστηκε η πιθανότητα επιρροής κάθε κόμβου. Στην συνέχεια, χρησιμοποιήθηκαν οι ακόλουθοι 4 μέθοδοι δημιουργίας node embeddings για την απεικόνιση των κόμβων σε χώρους με μικρότερες διαστάσεις:

1. Node2vec
2. DeepWalk
3. Walklets
4. Επέκταση της μεθόδου Graph2vec για κόμβους

Τέλος τα node embeddings που δημιουργήθηκαν, συνδυάστηκαν τις παρακάτω 6 μεθόδους ταξινόμησης για να γίνει η πρόβλεψη των ετικετών επιρροής των κόμβων.

1. GaussianNB
2. Decision Tree
3. K-nearest neighbors
4. Logistic Regression
5. Svm
6. Νευρωνικό δίκτυο

Κατόπιν εφαρμογής των παραπάνω μεθόδων σε 3 σύνολα δεδομένων από πραγματικά κοινωνικά δίκτυα προέκυψε ότι από τις προτεινόμενες μεθόδους, η πρόβλεψη της επιρροής μέσω συνδυασμού μηχανικής μάθησης και επέκτασης της μεθόδου Graph2vec για κόμβους παρουσιάζει φοβερή απόδοση για όλους τους αλγόριθμους ταξινόμησης που εξετάστηκαν. Από την άλλη η ταξινόμηση με την χρήση node embeddings των υπολοίπων μεθόδων παρουσιάζει ικανοποιητικά αποτελέσματα για μερικές από τις μεθόδους ταξινόμησης. Τέλος από τις μεθόδους ταξινόμησης που χρησιμοποιήθηκαν οι πιο αποτελεσματικές συνολικά για όλα τα πειράματα που πραγματοποιήθηκαν ήταν οι Svm και Νευρωνικό Δίκτυο.

## 5.2 Μελλοντική έρευνα

Σύμφωνα με την προτεινόμενη μεθοδολογία της παρούσας διπλωματικής εργασίας, για την αξιολόγηση της επιρροής των κόμβων ενός κοινωνικού δικτύου, λαμβάνεται υπόψη η πιθανότητα υιοθέτησης νέων ιδεών. Σε μελλοντική, έρευνα θα μπορούσαν να εξεταστούν νέοι τρόποι αξιολόγησης, οι οποίοι θα συνδέουν, για παράδειγμα, τοπολογικά χαρακτηριστικά του δικτύου με την πιθανότητα υιοθέτησης όπως υπολογίστηκε στην παρούσα εργασία. Παράλληλα, θα μπορούσε να αναζητηθεί ένας διαφορετικός τρόπος επιλογής παραμέτρων για τις μεθόδους δημιουργίας embeddings, που θα είναι αποτελεσματικότερος από την τυχαία αναζήτηση παραμέτρων. Τέλος, θα ήταν σπουδαία, η δημιουργία μιας αντίστοιχης μεθοδολογίας για την περίπτωση των δυναμικών κοινωνικών δικτύων. Τα δυναμικά κοινωνικά δίκτυα είναι δίκτυα στα όποια προστίθενται και αφαιρούνται κόμβοι και ακμές σε κάθε χρονικό διάστημα t όποτε η επιρροή κάθε κόμβου ίσως μεταβάλλεται με το πέρασμα του χρόνου. Έτσι η πρόβλεψη της επιρροής ενός κόμβου στον χρόνο t θα είχε πολλές εφαρμογές ουσιαστικές για την καθημερινότητα.

# Βιβλιογραφία

|  |  |
| --- | --- |
| [1] | S. Bhagat, G. Cormode και S. Muthukrishnan, «Node Classification in Social Networks,» 2011. |
| [2] | G. Zhao, P. Jia, C. Huang, A. Zhou και Y. Fang, «A Machine Learning Based Framework for Identifying Influential Nodes in Complex Networks,» 2020. |
| [3] | W. Kermack και A. G. McKendrick , «A contribution to the mathematical theory of epidemics,» 1927. |
| [4] | A. Filipovic, «Introduction to Node Embedding,» 2021. [Ηλεκτρονικό]. Available: https://memgraph.com/blog/introduction-to-node-embedding. |
| [5] | T. Mikolov, K. Chen, G. Corrado και J. Dean, «Efficient Estimation of Word Representations in Vector Space,» 2013. |
| [6] | P. Godec, «Graph Embeddings — The Summary,» 2018. [Ηλεκτρονικό]. Available: https://towardsdatascience.com/graph-embeddings-the-summary-cc6075aba007. |
| [7] | Q. V. Le και T. Mikolov, «Distributed Representations of Sentences and Documents,» 2014. |
| [8] | B. Perozzi, R. Al-Rfou και S. Skiena, DeepWalk: Online Learning of Social Representations, 2014. |
| [9] | A. Grover και J. Leskovec, «node2vec: Scalable Feature Learning for Networks,» 2016. |
| [10] | B. Perozzi, V. Kulkarni, H. Chen και S. Skiena, «Don't Walk, Skip! Online Learning of Multi-scale Network Embeddings,» 2016. |
| [11] | A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu και S. Jaiswal, «graph2vec: Learning Distributed Representations of Graphs,» 2017. |
| [12] | Wikipedia, «Machine Learning,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Machine\_learning. |
| [13] | P. Panzarasa, T. Opsahl και K. M. Carley, «Patterns and dynamics of users' behavior and interaction: Network analysis of an online community.,» 2009. |
| [14] | Stanford, «CollegeMsg dataset,» [Ηλεκτρονικό]. Available: https://snap.stanford.edu/data/CollegeMsg.html. |
| [15] | J. Leskovec και J. McAuley , «Learning to Discover Social Circles in Ego Networks,» 2012. |
| [16] | Stanford, «Facebook dataset,» [Ηλεκτρονικό]. Available: https://snap.stanford.edu/data/ego-Facebook.html. |
| [17] | J. Leskovec, D. Huttenlocher και J. Kleinberg, «Signed Networks in Social Media,» 2010. |
| [18] | Stanford, «Wiki-Vote dataset,» [Ηλεκτρονικό]. Available: https://snap.stanford.edu/data/wiki-Vote.html. |
| [19] | G. Rossetti, «Ndlib SI model,» 2018. [Ηλεκτρονικό]. Available: https://ndlib.readthedocs.io/en/latest/reference/models/epidemics/SIm.html. |
| [20] | Wikipedia, «Jenks natural breaks optimization,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Jenks\_natural\_breaks\_optimization. |
| [21] | R. Ahmad, «Jenks Natural Breaks — The Best Range Finder algorithm.,» [Ηλεκτρονικό]. Available: https://medium.com/analytics-vidhya/jenks-natural-breaks-best-range-finder-algorithm-8d1907192051. |
| [22] | B. Rozemberczki, O. Kiss και R. Sarkar, «Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs,» [Ηλεκτρονικό]. Available: http://karateclub.readthedocs.io/en/latest/index.html. |
| [23] | C. Martin και M. Riebeling, «A Process for the Evaluation of Node Embedding Methods in the Context of Node Classification,» 2020. |
| [24] | scikit learn, «Gaussian Naive Bayes,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/naive\_bayes.html. |
| [25] | scikit learn, «Decision Trees,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/tree.html. |
| [26] | scikit learn, «K-Nearest Neighbors,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/neighbors.html. |
| [27] | scikit learn, «Logistic Regresssion,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear\_model.LogisticRegression.html. |
| [28] | scikit learn, «SVM,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/svm.html. |
| [29] | Wikipedia, «Artificial Neural Network,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Artificial\_neural\_network. |
| [30] | Keras, «Hyperband Tuner,» [Ηλεκτρονικό]. Available: https://keras.io/api/keras\_tuner/tuners/hyperband/. |
| [31] | L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh και A. Talwalkar, «Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization,» 2016. |
| [32] | Wikipedia, «Cross-validation,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Cross-validation\_(statistics). |
| [33] | scikit learn, «K-Fold cross validation,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model\_selection.KFold.html. |
| [34] | A. Bhandari, «Everything you Should Know about Confusion Matrix for Machine Learning,» [Ηλεκτρονικό]. Available: https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/. |
| [35] | T. Kajdanowicz, P. Kazienko και P. Doskocz, «Label-Dependent Feature Extraction in Social Networks for Node Classification,» 2010. |
| [36] | A. Anagnostopoulos, R. Kumar και Μ. Mahdian, «Influence and correlation in social networks,» 2008. |
| [37] | F. Chen, Y.-C. Wang, C.-C. J. Kuo και B. Wang, «Graph representation learning: a survey. Cambridge University Press,» 2020. |
| [38] | J. Heidemann, M. Klier και F. Probst, «Identifying key users in online social networks: A pagerank based approach,» 2010. |
| [39] | M. Grohe, «word2vec, node2vec, graph2vec, X2vec: Towards a Theory of Vector Embeddings of Structured Data. », 2020 |
| [40] | M. Lei, L. Zhi και T. Xiang-Yang, "Evaluating influential spreaders in complex networks by extension of degree" 2015 |
| [41] | B. Hou, Y. Yao και D. Liao, "Identifying all-around nodes for spreading dynamics in complex networks" 2012 |
| [42] | J. Leskovec, J. Kleinberg και C. Faloutsos, "Graph evolution: Densification and shrinking diameters" 2007 |
| [43] | D. Bucur, "Top influencers can be identified universally by combining classical centralities" 2020 |
| [44] | N. Agarwal, H. Liu, L. Tang και P. Yu, " Identifying the influential bloggers in a community " 2008 |
| [45] | J. Bondy και U. Murty, " Graph Theory with Applications " 1976 |
| [46] | F. D. Malliaros, M.-E. G. Rossi και M. Vazirgiannis " Locating influential nodes in complex networks. " 2016 |