ΠΑΡΟΥΣΙΑΣΗ ΠΤΥΧΙΑΚΗΣ “ΠΡΟΓΝΩΣΗΣ ΑΚΜΩΝ ΣΕ ΨΗΦΙΑΚΑ ΚΟΙΝΩΝΙΚΑ ΔΥΚΤΙΑ ΜΕ ΧΡΗΣΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ”

Τι ειναι ένα κοινωνικό δίκτυο?

Ένα κοινωνικό δίκτυο είναι μια κοινωνική δομή αποτελούμενη από άτομα (ή οργανισμούς) που ονομάζονται "κόμβοι", οι οποίοι συνδέονται με έναν ή περισσότερους συγκεκριμένους τύπους αλληλεξάρτησης, όπως η φιλία, η συγγένεια, το κοινό συμφέρον, η οικονομική ανταλλαγή, σχέσεις πεποιθήσεων, γνώσης ή γοήτρου.

Υπάρχουν πολλά κοινωνικά δίκτυα όπως το facebook,twitter,instagram,linkedin,youtube ή το stackoverflow.

Γιατί κοινωνικά δίκτυα και όχι κάποια άλλη διαδικτυακή υπηρεσία?

Αυτό που κάνει τα online κοινωνικά δίκτυα να ξεχωρίζουν από τις υπόλοιπες διαδικτυακές υπηρεσίες είναι:

- Τα εξελιγμένα εργαλεία που επιτρέπουν στους χρήστες να διαμοιράζονται ψηφιακά αρχεία (π.χ. κείμενο, εικόνες και άλλα)
- Τα εξελιγμένα εργαλεία για την επικοινωνία και την κοινωνικοποίηση των χρηστών

Διατύπωση του προβλήματος της πρόγνωσης ακμών

Τα δεδομένα του δικτύου μπορούν να διατυπωθούν ως εξής:

$$e_{ij}(t)=(v_i,v_j,t)(1) \quad t_{min}\le t\le t_{max}$$

Όπου t_{min} , t_{max} είναι παλαιότερη και νεότερη χρονική παρατήρηση μέσα στο σύνολο των διαθέσιμων δεδομένων. V_i,V_j είναι οι κόμβοι που συνδέονται.

Το συγκεκριμένο χρονικό διάστημα $T=[t_{min},t_{max}]$ διαμερίζεται σε ένα σύνολο N μη επικαλυπτόμενων χρονικών περιόδων $T_1,T_2,\dots T_J,\dots T_N$ ίσης χρονικής διάρκειας δt θεωρώντας ένα σύνολο N+1 χρονικών στιγμών $t_j=t_{0,}+j*\delta t(2)\ 1\le j\le N$ $t_0,t_1,t_2,\dots t_{j-1},t_j,\dots,t_{N-1},t_N$ τέτοιων ώστε

$$\delta t=\frac{\Delta T}{N}(4)$$

Όπου ΔT = t_{max}-t_{min} και $$T_j=\begin{cases}[t_{j-1},t_j),1\le j\le N-1;\\ [t_{j-1},t_j],j=N.\end{cases}(5)$$

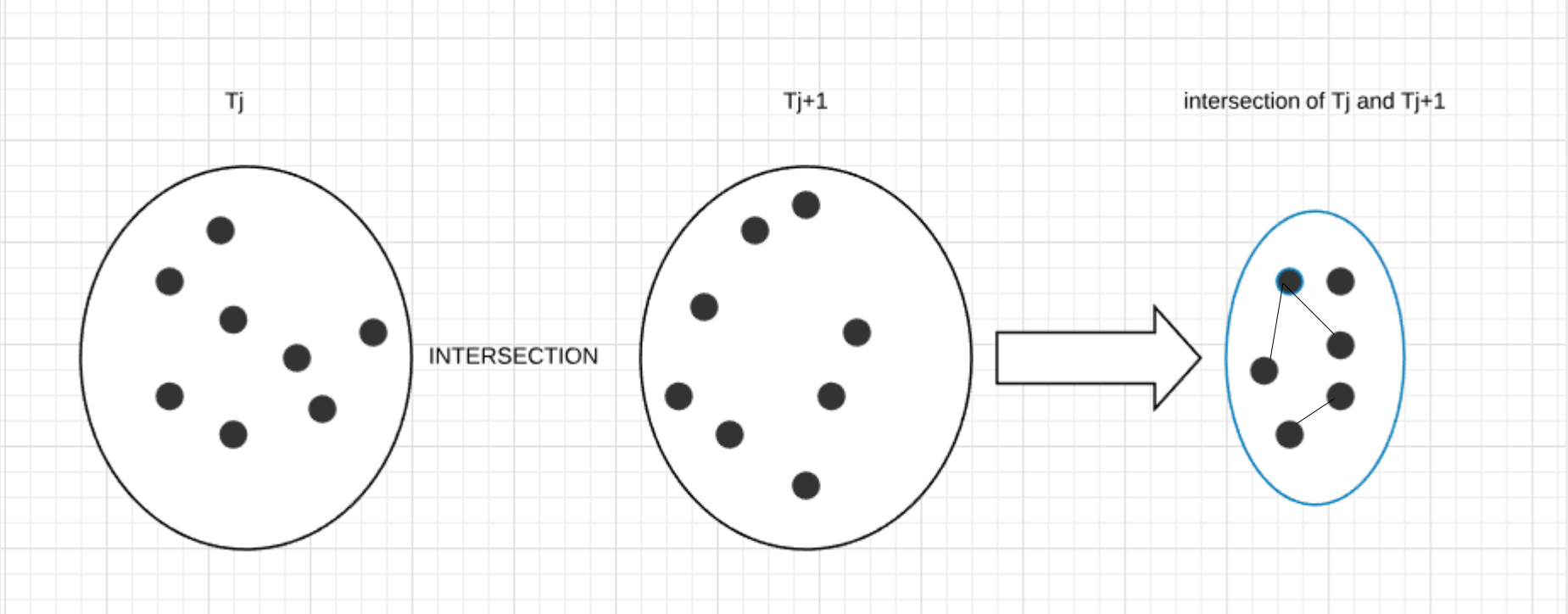
$G[t_{j-1},t_j]=(V[t_{j-1},t_j],E[t_{j-1},t_j])$

Ο ορισμός της j-οστής χρονικης στιγμης

Για κάθε $V[t_{j-1},t_j]$ από τις χρονικές στιγμές μπορούμε να θεωρήσουμε το αντίστοιχο υπο-γράφημα του συνολικού δικτύου: $E[t_{j-1},t_j]$.

Όπου είναι το σύνολο των κορυφών που εμφανίζονται στα άκρα των ακμών

...ΣΥΝΕΧΕΙΑ ΔΙΑΤΥΠΩΣΗΣ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ (2)



Μας ενδιαφέρουν μόνο οι ακμές της τομής των 2 γραφημάτων.
Το γράφημα της τομής το συμβολίζουμε ως εξής $V^*[t_{j-1},t_{j+1}]$

Επίσης, μας ενδιαφέρει ο διαχωρισμός των ακμών στο γράφημα στις ακμές που δημιουργήθηκαν την αμέσως προηγούμενη χρονική στιγμη και την αμέσως επόμενη.

$$E^{\downarrow}[t_{j-1},t_j]=\{(u,v)\in E[t_{j-1},t_j]: u\in V^{\downarrow}[t_{j-1},t_{j+1}] \text{ και } v\in V^{\downarrow}[t_{j-1},t_{j+1}]\} \quad (7)$$

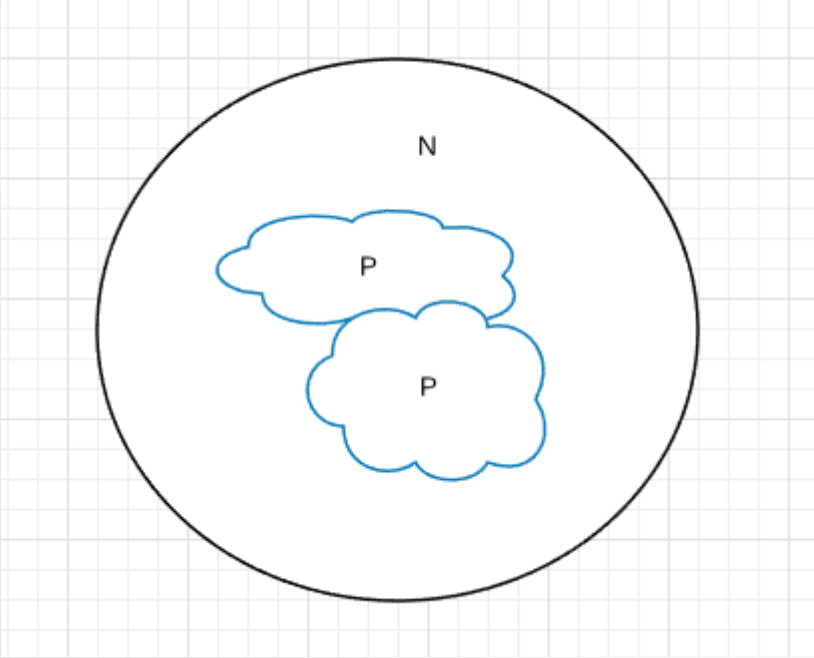
$$E^{\downarrow}[t_j,t_{j+1}]=\{(u,v)\in E[t_j,t_{j+1}]: u\in V^{\downarrow}[t_{j-1},t_{j+1}] \text{ και } v\in V^{\downarrow}[t_{j-1},t_{j+1}]\} \quad (8)$$

....ΣΥΝΕΧΕΙΑ ΔΙΑΤΥΠΩΣΗΣ ΠΡΟΒΛΗΜΑΤΟΣ (3)

Επίσης έχουμε και το γράφημα N (negative) που δηλώνει τις ακμές οι οποίες δε δημιουργήθηκαν. Το N εκφράζεται ως εξής : N = ((V*) * (V*)/E_after)/E_before

Δηλαδή είναι το καρτεσιανό γινόμενο του V* αφαιρώντας όμως τους δεσμούς που έχουν δημιουργηθεί.

Στο παρακάτω σχήμα το N δείχνει το σύνολο negative και το P το σύνολο Positive, ηθεί πριν και μετά.



....ΣΥΝΕΧΕΙΑ ΔΙΑΤΥΠΩΣΗΣ ΠΡΟΒΛΗΜΑΤΟΣ (4)

Σκοπός μας είναι να προβλέψουμε τις ακμές που δημιουργούνται στην επόμενη χρονική στιγμή, χρησιμοποιώντας trainining στο N και στο $E^*[t_{j-1},t_j]$ και testing στο $E^*[t_j,t_{j+1}]$

Για να πραγματοποιήσουμε το παραπάνω πρέπει να χρησιμοποιήσουμε κάποιες μετρικές μεθόδους. Αυτές είναι οι παρακάτω:

i. $S_{GD}=[S_{GD}(u,v)]=-Length\ of\ Shortest\ Path\ Between u \wedge v$ **[Graph Distance]**

ii. $S_{CN}=[S_{CN}(u,v)]=¿\Gamma(u)\cap\Gamma(v)\vee¿$ **[Common Neighbors]** όπου $\Gamma(u)$ το σύνολο των γειτόνων του κόμβου u.

iii. $S_{JC}=[S_{JC}(u,v)]=¿\Gamma(u)\cap\Gamma(v)\vee\frac{¿}{¿\Gamma(u)\cup\Gamma(v)\vee¿¿}¿$ **[Jaccard's Coefficient]**

iv. $S_A=[S_A(u,v)]=\sum_{z\in\Gamma(u)\cap\Gamma(v)}\frac{1}{\log ¿¿}¿$ **[Adamic / Adar]**

v. $S_{PA}=[S_{PA}(u,v)]=¿\Gamma(u)\vee¿\vee\Gamma(v)\vee¿$ **[Preferential Attachment]**

Τα παραπάνω υπολογίζονται για κάθε ζεύγος κόμβων στα γραφήματα N και E*before

...ΣΥΝΕΧΕΙΑ ΔΙΑΤΥΠΩΣΗΣ ΠΡΟΒΛΗΜΑΤΟΣ (5)

Κάνοντας τις μετρήσεις αυτές λοιπόν θα έχουμε μια μεγάλη λίστα με μετρήσεις κόμβων από τα γραφήματα N και E_before και μια ακόμη παράλληλη λίστα Y με τα αποτελέσματα που θα δείχνει αν ο κόμβος δημιουργήθηκε(=1) ή όχι (=0)

Η κάθε γραμμή δηλώνει τις 5 μετρήσεις του εκάστοτε ζευγαριού κόμβων

U,v

F1	F2	F3	F4	F5

0
1
0
1
1
1
0
0
1
1

ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ

Το σύνολο δεδομένων που θα υλοποιήσουμε τον αλγόριθμο είναι το STACKOVERFLOW TEMPORAL NETWORK DATASET (https://snap.stanford.edu/data/sx-stackoverflow.txt.gz)

Dataset statistics (sx-stackoverflow)	
Nodes	2601977
Temporal Edges	63497050
Edges in static graph	36233450
Time span	2774 days

Η εικόνα μας δίνει πληροφορίες, όπως ότι οι κόμβοι ειναι 2601977, ενώ οι ακμές του γραφήματος ειναι 63 εκατομμύρια περίπου και ότι αυτό δημιουργήθηκε σε χρονικό διάστημα 7,5 χρόνων. Κάθε γραμμή έχει 3 στήλες. Οι 2 πρώτες είναι οι κόμβοι που αλληλεπέδρασαν, ενώ η τρίτη είναι το χρονικό στιγμιότυπο.Επομένως μια ακμή καθορίζεται ως εξής (u,v,t).

```
9 8 1217567877
1 1 1217573801
13 1 1217606247
17 1 1217617639
48 2 1217618182
17 1 1217618239
19 9 1217618357
```

΄Οπως φαίνεται η κάθε ακμή του δικτύου είναι συσχετισμένη με μια χρονοσ-φραγίδα, που δείχνει τη χρονική στιγμή δημιουργίας της. Κάθε ακμή αποτελείται απο τα στοιχεία (source_id, target_id,timestamp), όπου το source_id είναι το αναγνωριστικό του κόμβου προέλευσης της ακμής, το target_id είναι το αναγνωριστικό του κόμβου κατάληξης της ακμής ενώ το timestamp υποδηλώνει την χρονική στιγμή δημιουργίας της ακμής.

Ποιες είναι οι αλληλεπιδράσεις που δημιουργούν το σύνολο δεδομένων:

- ο χρήστης u απάντησε στο ερώτημα του χρήστη v κατά το χρόνο t
- ο χρήστης u σχολίασε την ερώτηση του χρήστη v κατά το χρόνο t
- ο χρήστης u σχολίασε την απάντηση του χρήστη v κατά το χρόνο t

TRAIN-TEST SPLIT ~ ΧΡΗΣΗ ΤΟΥ K-fold CROSS VALIDATION ΓΙΑ K = 10

P U N -> δηλώνει την ένωση του Positive , Negative.

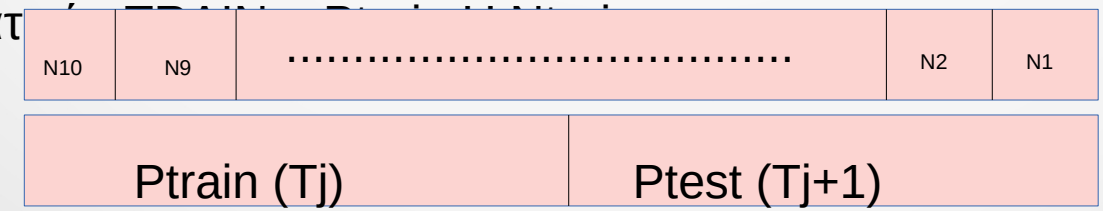
P = $E^*[t_{j-1},t_j]$ U $E^*[t_j,t_{j+1}]$ όπου το $E^*[t_{j-1},t_j]$ θα το λέμε εν συντομία Ptrain και το $E^*[t_j,t_{j+1}]$ θα το λέμε Ptest, άρα

Ptrain = $E^*[t_{j-1},t_j]$
Ptest = $E^*[t_j,t_{j+1}]$

Το σύνολο N θα το διαμερίσουμε σε 10 folds με τον αλγόριθμο k-fold cross validation.

Οπότε, θα έχουμε: N = N(f1) u N(f2) u N(f3) u N(f4) u N(f5) u N(f6) u N(f7) u N(f8) u N(f9) u N(f10)

Συμπερασματ



Τα fold διατυπώνονται ως εξής f1: TRAIN = [N2,N3,..,N10,Ptrain] , TEST=[N1,Ptest]
f2: TRAIN = [N1,N3,..,N10,Ptrain] , TEST=[N2,Ptest]
κ.ο.κ.

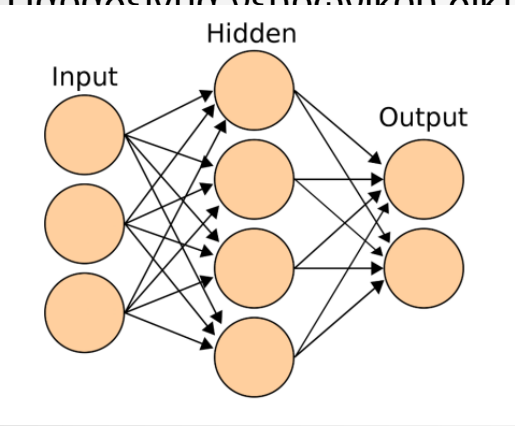
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΠΡΟΓΝΩΣΗ ΜΕ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Γιατί χρήση νευρωνικών δικτύων:

Το πλεονέκτημα της χρήσης των νευρωνικών δικτύων για πρόβλεψη είναι ότι μπορούν να μάθουν μόνο από παραδείγματα και ότι μετά την ολοκλήρωση της εκμάθησης τους, είναι σε θέση να πιάσουν κρυφές και έντονες μη γραμμικές εξαρτήσεις, ακόμη και όταν υπάρχει σημαντικός θόρυβος στο training σετ.

Στην εργασία γίνεται χρήση νευρωνικού δικτύου με είσοδο 5 νευρώνες λόγω 5 χαρακτηριστικών και έξοδο 1 νευρώνα που δείχνει 0 ή 1. Το hidden layer το δηλώσαμε εμείς κάνοντας διάφορα παραδείγματα,ώστε να ειναι και μικρό για ταχύτητα και γρήγορο. Έχουμε, δηλαδή 2 κρυφά επίπεδα το 1ο με 45 νευρώνες και το δεύτερο με 90

Παράδειγμα νευρωνικού δικτύου σε φωτογραφία:



ΠΑΡΑΔΕΙΓΜΑ ΑΠΟΤΕΛΕΣΜΑΤΟΣ ΑΠΟ ΕΝΑ FOLD ΚΑΙ ΤΕΛΙΚΟ ACCURACY

– Fold 2

```
Accuracy Score Of Fold:  0.9560772579933636
Classification Report:                precision    recall  f1-score   support

    class 0       1.00      0.96      0.98     51200
    class 1       0.29      1.00      0.45       937

avg / total       0.99      0.96      0.97     52137

Confusion Matrix Follows:  [[48911  2289]
 [    1   936]]
```

Αποτελέσματα του 2ου fold από τα 10 με Classification Report και Confusion Matrix

ΤΕΛΙΚΟ ACCURACY ~ ΜΕΣΟΣ ΟΡΟΣ ΤΩΝ ACCURACY ΑΠΟ ΤΑ 10 FOLD

```
Accuracy for T 1 0.9554743972476711
```

ΤΕΛΟΣ ΠΑΡΟΥΣΙΑΣΗΣ

Ευχαριστώ για το χρόνο σας.

Ζερκελίδης Δημήτριος