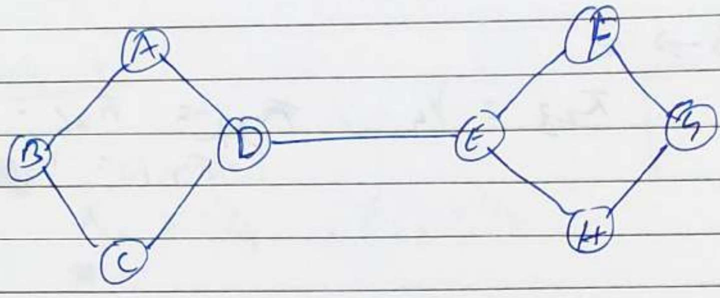
Q1 Node2Vec

1

1)



[NODE2VEC Implementation]

Node2vec :-

~~Note~~ Transition probability is given by

$$\pi_{uv} = \begin{cases} 1/p & w_{uv} \\ 1/q & w_{uv} \\ 1 & w_{uv} \end{cases}$$

We assume all nodes to have a uniform weight ~~of 1~~. of 1.

$p = 4, \quad q = 2$

(I) Determine the walks of every node.

Node A

$A \rightarrow D$

$$\pi_{DA} = 1/p = 1/4$$

$$\pi_{DE} = 1/q = 1/2$$

$$\pi_{DC} = 1/q = 1/2$$

$A \rightarrow D \rightarrow E$

Node B

$B \rightarrow A$

$\pi_{AB} = 1/4$, $\pi_{AD} = 1/2$

$B \rightarrow A \rightarrow D$

Node C

$C \rightarrow B$

$\pi_{BC} = 1/4$, $\pi_{BA} = 1/2$

$C \rightarrow B \rightarrow A$

Node D

$D \rightarrow E$

$\pi_{ED} = 1/4$, $\pi_{EF} = 1/2$, $\pi_{EH} = 1/2$

$D \rightarrow E \rightarrow H$

Node E

$E \rightarrow D$

$\pi_{DE} = 1/4$, $\pi_{DC} = 1/2$, $\pi_{DA} = 1/2$

$E \rightarrow D \rightarrow A$

Node F

$F \rightarrow G$

$\pi_{GF} = 1/4, \quad \pi_{GH} = 1/2$

$F \rightarrow G \rightarrow H$

Node G

$G \rightarrow H$

$\pi_{HG} = 1/4, \quad \pi_{HE} = 1/2$

$G \rightarrow H \rightarrow E$

Node H

$H \rightarrow E$

$\pi_{ED} = 1/2, \quad \pi_{EH} = 1/4, \quad \pi_{EF} = 1/4$

$H \rightarrow E \rightarrow D$

The Random walks are

$A \rightarrow D \rightarrow E$	$G \rightarrow H \rightarrow E$
$B \rightarrow A \rightarrow D$	$H \rightarrow E \rightarrow D$
$C \rightarrow B \rightarrow A$	
$D \rightarrow E \rightarrow H$	
$E \rightarrow D \rightarrow A$	
$F \rightarrow G \rightarrow H$	

$$L = -\sum_{u \in V} \sum_{v \in N_R(u)} \log \left(\frac{\exp(z_u^T z_v)}{\sum_{n \in V} \exp(z_u^T z_n)} \right)$$

$$\frac{\partial L}{\partial z_u} = \frac{1}{\exp(z_u^T z_v)} \cdot \exp(z_u^T z_v) \cdot z_v$$

$$- \frac{1}{\sum \exp(z_u^T z_n)} \sum \exp(z_u^T z_n) z_n$$

$$\frac{\partial L}{\partial z_u} = -\sum_{v \in N_R(u)} z_v - \frac{\sum e^{z_u^T z_n} z_n}{\sum e^{z_u^T z_n}}$$

Initilization :-

$$\begin{array}{c} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{array} \begin{bmatrix} 0.2 & 0.3 \\ 0.4 & 0.1 \\ 0.3 & 0.2 \\ 0.25 & 0.4 \\ 0.3 & 0.7 \\ 0.6 & 0.8 \\ 0.3 & 0.9 \\ 0.7 & 0.4 \end{bmatrix}$$

Determine Gradients:-

Node A :-

$$\sum e^{z_A^T z_u} = 8.86$$

$$\sum e^{z_u^T z_v} z_u = [0.907 , 0.518]$$

$$\left.\frac{\partial L}{\partial z_A}\right|_{v=D} = [0.25 , 0.4] - \frac{[0.907 , 0.518]}{8.86}$$

$$\left.\frac{\partial L}{\partial z_A}\right|_{v=D} = [0.1476 , 0.3415]$$

$$\left.\frac{\partial L}{\partial z_A}\right|_{v=E} = [0.3 , 0.7] - \frac{[0.907 , 0.518]}{8.86}$$

$$= [0.197 , 0.64]$$

$$\frac{\partial L}{\partial z_A} = \left.\frac{\partial L}{\partial z_A}\right|_{v=D} + \left.\frac{\partial L}{\partial z_A}\right|_{v=E}$$

$$\frac{\partial L}{\partial z_A} = [0.344 , 0.9830]$$

Node B :-

$$\sum e^{z_A^T z_n} = 8.61$$

$$\sum e^{z_B^T z_n} z_n = [0.963, 0.550]$$

$$\frac{\partial L}{\partial z_B}\Big|_{v=A} = [0.2, 0.3] - \frac{[0.963, 0.550]}{8.61}$$

$$= [0.088, 0.23]$$

$$\frac{\partial L}{\partial z_B}\Big|_{v=D} = [0.25, 0.4] - \frac{[0.963, 0.550]}{8.61}$$

$$= [0.138, 0.336]$$

$$\frac{\partial L}{\partial z_B} = \frac{\partial L}{\partial z_B}\Big|_{v=A} + \frac{\partial L}{\partial z_B}\Big|_{v=D}$$

$$\frac{\partial L}{\partial z_B} = [0.22, \underline{\underline{0.57}}]$$

Node C :-

$$\sum e^{z_C^T z_n} = 8.25$$

$$\sum e^{z_C^T z_n} z_n = [0.7354, 0.53]$$

$$\frac{\partial L}{\partial z_C}\Big|_{v=B} = [\cancel{0.3, 0.2}] - [0.29, 0.03]$$

$$\left.\frac{\partial L}{\partial z_C}\right|_{v=A} = [0.09, 0.23]$$

$$= [0.38, 0.26]$$

Node D

$$\sum e^{z_D^T z_u} = 9.47$$

$$\sum e^{z_D^T z_u} z_u = [0.97, 0.55]$$

$$\left.\frac{\partial L}{\partial z_D}\right|_{v=E} = [0.196, 0.64]$$

$$\left.\frac{\partial L}{\partial z_D}\right|_{v=H} = [0.5916, 0.34]$$

$$\frac{\partial L}{\partial z_D} = [0.786, 0.98]$$

Node E

$$\sum e^{z_E^T z_u} = 10.98$$

$$\sum z_e^{T z_u} z_u = [1.142, 0.625]$$

$$\left.\frac{\partial L}{\partial z_E}\right|_{v=D} = [0.145, 0.34]$$

8

$$\left.\frac{\partial L}{\partial z_E}\right|_{v=H} = [0.09, \ 0.24]$$

$$\frac{\partial L}{\partial z_E} = [0.241, \ 0.581]$$

Node F

$$\sum e^{z_F^T z_u} = 12.42$$

$$\sum e^{z_F^T z_u} z_u = [1.46, \ 0.83]$$

$$\left.\frac{\partial L}{\partial z_F}\right|_{v=G} = [0.18, \ 0.83]$$

$$\left.\frac{\partial L}{\partial z_F}\right|_{v=H} = [0.58, \ 0.33]$$

$$\frac{\partial L}{\partial z_F} = [0.76, \ 1.16]$$

Node G

$$\sum e^{z_G^T z_u} = 11.76$$

$$\sum e^{z_G^T z_u} z_u = [1.23, \ 0.707]$$

$$\frac{\partial L}{\partial z_H} = [0.59, \ 0.33]$$

$$\left.\frac{\partial L}{\partial z_H}\right| = [0.194, \ 0.63]$$

$$\frac{\partial L}{\partial z_H} = [0.78, \ 0.97]$$

Node H :-

$$\sum e^{z_H^T z_u} = 10.9$$

$$\sum e^{z_H^T z_u} z_u = [0.53, 1.59]$$

$$\left.\frac{\partial L}{\partial z_H}\right|_{v=E} = [0.25, 0.55]$$

$$\left.\frac{\partial L}{\partial z_H}\right|_{v=D} = [0.201, 0.25]$$

$$\frac{\partial L}{\partial z_H} = [0.45, 0.80]$$

Update Embeddings

$$\eta = 0.1$$

$$\rightarrow \begin{bmatrix} 0.2 & 0.3 \\ 0.4 & 0.1 \\ 0.3 & 0.2 \\ 0.25 & 0.4 \\ 0.3 & 0.7 \\ 0.6 & 0.8 \\ 0.3 & 0.9 \\ 0.7 & 0.4 \end{bmatrix} - (-0.1) \begin{bmatrix} 0.22 & 0.57 \\ 0.79 & 0.03 \\ 0.38 & 0.76 \\ 0.786 & 0.98 \\ 0.241 & 0.587 \\ 0.76 & 1.16 \\ 0.78 & 0.97 \\ 0.45 & 0.8 \end{bmatrix}$$

Iteration 2

→ Using the same process as Explained in iteration 1 we get the gradients for iteration 2.

Node A $\dfrac{\partial L}{\partial z_A} = [0.25, 0.85]$

Node B

$\dfrac{\partial L}{\partial z_B} = [0.14, 0.44]$

Node C

$\dfrac{\partial L}{\partial z_C} = [0.35, 0.74]$

Node D

$$\frac{\partial L}{\partial z_D} = [0.74, 0.86]$$

Node E

$$\frac{\partial L}{\partial z_E} = [0.155, 0.40]$$

Node F

$$\frac{\partial L}{\partial z_F} = [0.16, 1.01]$$

Node G

$$\frac{\partial L}{\partial z_G} = [0.73, 0.86]$$

Node H

$$\frac{\partial L}{\partial z_H} = [0.38, 0.69]$$

$$\hat{z}_4 = z_4 - \eta \frac{\partial L}{\partial z_4}$$

$$= \begin{bmatrix} 0.265 & 0.46 \\ 0.46 & 0.77 \\ 0.38 & 0.25 \\ 0.41 & 0.60 \\ 0.37 & 0.80 \\ 0.76 & 1.04 \\ 0.46 & 1.16 \\ 0.79 & 0.57 \end{bmatrix}$$

Q1 Deep walk

13

DeepWalk [DeepWalk Implementation]

The Random Walk for ~~the~~ Deep Walk is as show in Table below

$A \to B \to C$
$B \to C \to D$
$C \to D \to A$
$D \to E \to F$
$E \to B \to A$
$F \to G \to H$
$G \to F \to E$
$H \to E \to D$

$$L = -\sum_{u \in V} \sum_{v \in N_R(u)} \log \left(\frac{e^{z_u^T z_v}}{\sum_{n \in V} \exp(z_u^T z_v)} \right)$$

$$\frac{\partial L}{\partial z_u} = -\sum z_v - \frac{\sum e^{z_u^T z_v} z_n}{\sum e^{z_u^T z_v}}$$

D.

Initialization :-

$$\begin{array}{c} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{array} \begin{bmatrix} 0.2 & 0.3 \\ 0.4 & 0.1 \\ 0.3 & 0.2 \\ 0.25 & 0.4 \\ 0.3 & 0.7 \\ 0.6 & 0.8 \\ 0.3 & 0.9 \\ 0.7 & 0.4 \end{bmatrix}$$

Iteration 1

We use the same process as described for Node2Vec.

The ~~a~~ obtained gradients are

Node A.

$$\frac{\partial L}{\partial z_A} = [0.49, 0.18]$$

Node B

$$\frac{\partial L}{\partial z_B} = [0.32, 0.47]$$

Node C

$[0.23, 0.57] = \frac{\partial L}{\partial z_C}$

Node D

$\frac{\partial L}{\partial z_D} = [0.69, 1.38]$

Node E

$\frac{\partial L}{\partial z_E} = [0.29, 0.58]$

Node F

$\frac{\partial L}{\partial z_F} = [0.76, 1.16]$

Node G

$\frac{\partial L}{\partial z_G} = [0.68, 1.37]$.

Node H

$\frac{\partial L}{\partial z_H} = [0.45, 0.80]$

Date ______
Page ______

update node Embeddings

$$z_u = z_u - \eta \frac{\partial L}{\partial z_u} \qquad \eta = 0.1$$

$$z_u = \begin{bmatrix} 0.298 & 0.336 \\ 0.464 & 0.194 \\ 0.346 & 0.314 \\ 0.388 & 0.676 \\ 0.348 & 0.816 \\ 0.252 & 1.032 \\ 0.436 & 1.174 \\ 0.79 & 0.56 \end{bmatrix}$$

Iteration 2

Node A $\partial L/\partial z_A = [0.5[illegible]3, 0.26]$

Node B $\partial L/\partial z_B = [0.40, 0.64]$

Node C $\partial L/\partial z_C = [0.34, 0.71]$

Node D $\partial L/\partial z_D = [0.77, 1.53]$

E $\partial L/z_E = [0.34, 0.71]$

F $\partial L/\partial z_F = [0.86, 1.35]$

G $\partial L/\partial z_G = [0.77, 1.59]$

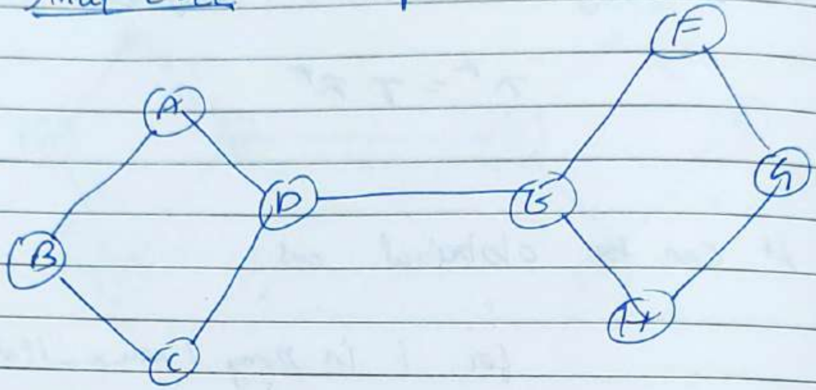
H $\partial L/\partial z_H = [0.52, 0.71]$

update ~~weight~~ embedding vector

$$= \begin{bmatrix} 0.362 & 0.344 \\ 0.472 & 0.211 \\ 0.357 & 0.328 \\ 0.396 & 0.691 \\ 0.358 & 0.829 \\ 0.464 & 1.051 \\ 0.445 & 1.196 \\ 0.797 & 0.571 \end{bmatrix}$$

[Struct2Vec]

Strut 2vec implementation



$$f_k(u,v) = f_{k-1}(u,v) + g(s(R_k(u)), s(R_k(v)))$$

$$g(b_1, b_2) = \frac{\max(b_1, b_2)}{\min(b_1, b_2)} - 1$$

The Diameter of a Graph is $k^* = 5$ but for simplicity we choose $k^* = 2$.

Strut2veck has mainly 4 steps

1) → Determine Structural Similarity between vertex pairs

2) → Construct weighted multilayer Graph

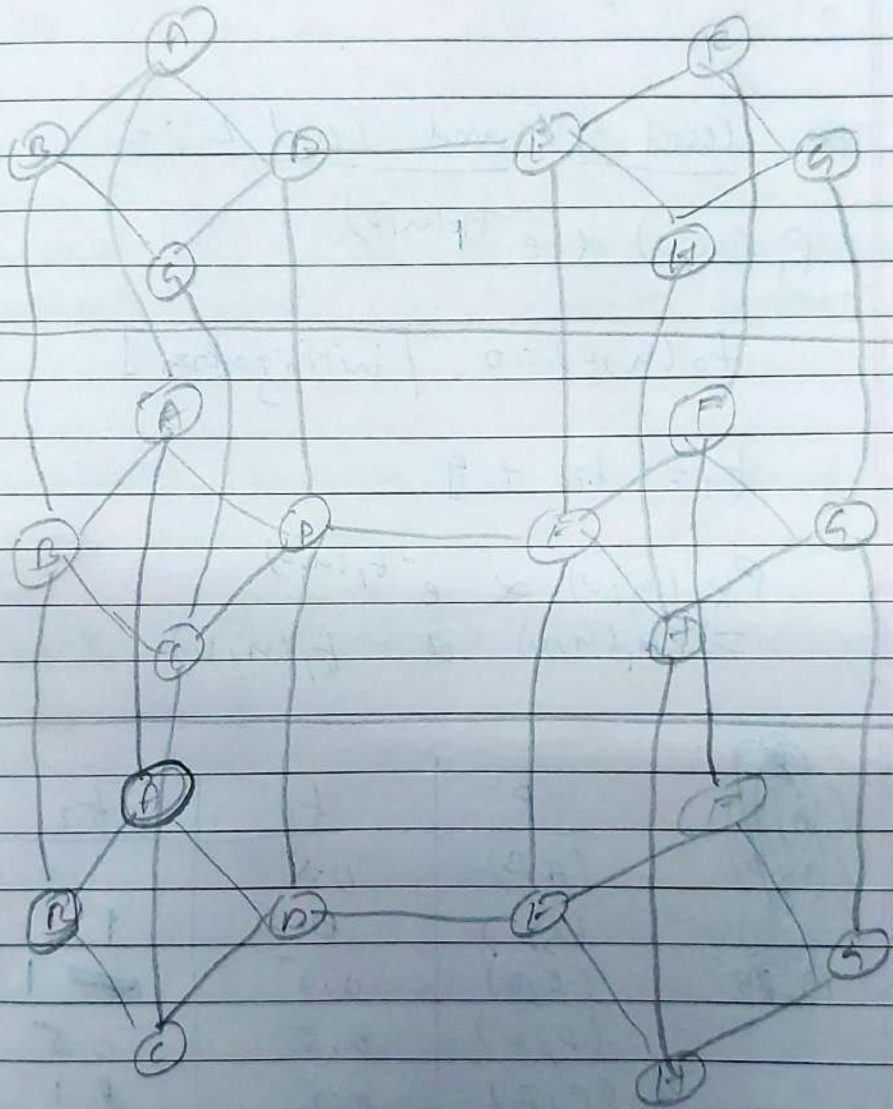
3) → use multilayer Graph to Generate context

4) → Apply Deepwalk (on any other Standard method) to obtain embeddings

$K=2$

$K=1$

$K=0$



Level 0 :-

The probability of choosing a node v given u for level 0 is 1. for all nodes.

level 1 and level 2 :-

$$P_1(u,v) \propto e^{-t_1(u,v)}$$

$$t_0(u,v) = 0 \quad [\text{initialization}]$$

$$t_1 = t_0 + g$$

$$P_K(u,v) \propto e^{-t_1(u,v)}$$
$$\Rightarrow P_1(u,v) \propto -t_1(u,v)$$

P	t_1	t_2
(A,B)	0.5	1
(B,C)	0.5	1
(C,D)	0.5	1
(D,E)	0.5	0.5
(E,F)	0.5	1
(F,G)	0.5	1
(G,H)	0.5	1
(A,D)	0.5	1
(H,E)	0.5	1

The interlayer transition probability is given ~~using~~ by the formula.

$$w(u_k, u_{k+1}) = \log(\Gamma_k(u) + e)$$

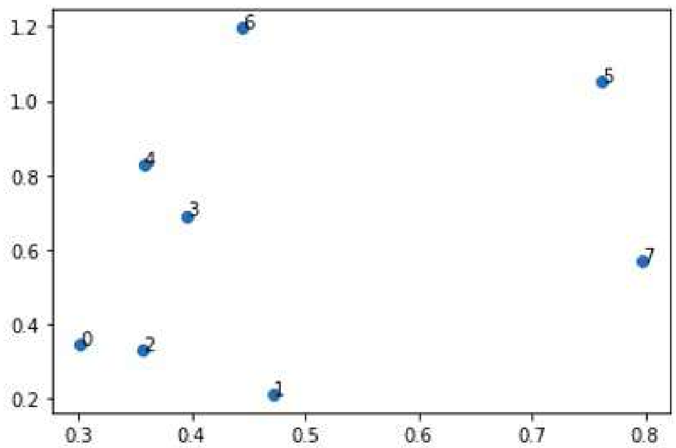
$$w(u_k, u_{k+1}) \propto \Gamma_k(u)$$

where Γ is the similarity function between node u and other nodes in k.

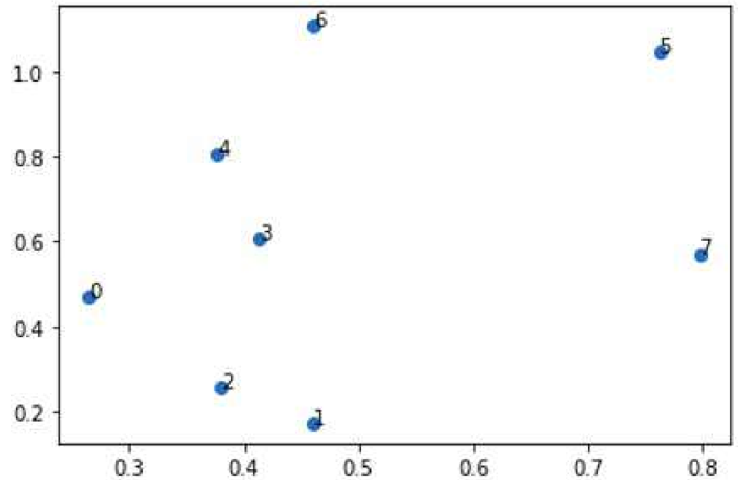
The walks generated by Struct2Vec are ~~using~~ walk_length = 4.

~~$N_R(u) = \{A, D, C\} \cup \{A, D, E\}$~~

$$N_R(u) = \begin{bmatrix} \{A, D, E, F\} \cup \{A, D, C, B\} \\ \{B, C, D, E\} \cup \{B, A, D, E\} \\ \{C, D, E, F\} \cup \{C, B, A, D\} \\ \{D, E, F, G\} \cup \{D, A, B, C\} \\ \{E, D, A, B\} \cup \{E, F, G, H\} \\ \{F, E, D, A\} \cup \{F, G, H, E\} \\ \{G, H, E, D\} \cup \{G, F, E, D\} \\ \{H, E, D, A\} \cup \{H, E, F, G\} \end{bmatrix}$$



DeepWalk



Node2Vec

Apply these random walks to
deepwalk or node2vec and
Generate embeddings.

1.2) The result of the Embedding is shown
in the ~~above table~~. figure

→ It can be observed that both node2vec
and Deepwalk have resulted
in similar embeding

→ the nodes D, E (3, 4) are
close to each other as expected
and A, B, C (0, 1, 2) and
F, G, H (5, 6, 7) are on
opposite sides.

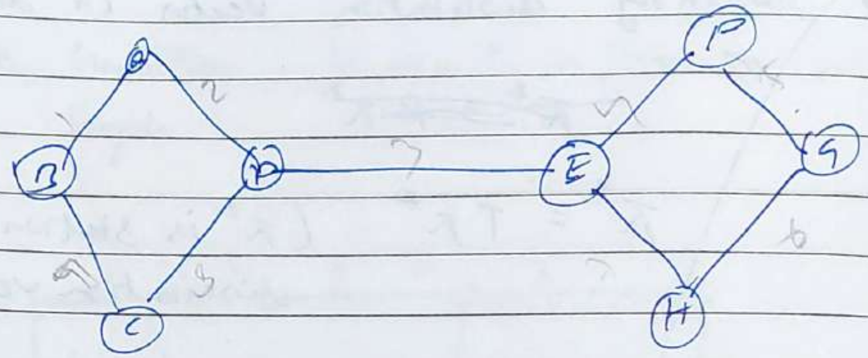
→ Deepwalk :- performs random walks
and more frequently the
a given node appears in a ~~ran~~ random
walk likely they are similar.

Node2vec :- It uses its memory to Generate
the sequence of paths to be ~~tat~~ taken.
it is an improved form of
Deepwalk.

Struct2vec :- It captures structural information across hops.

-> In this case since the graph is small (8 nodes), the paths traversal by both Deepwalk and node2vec ~~are similar hence they~~ are similar hence they have generated similar results.

1.3)



The transition matrix T is given by

$$T = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{array} \begin{array}{c} \begin{array}{cccccccc} A & B & C & D & E & F & G & H \end{array} \\ \begin{bmatrix} 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 1/3 & 0 & 1/3 & 0 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/3 & 0 & 1/3 & 0 & 1/3 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \end{bmatrix} \end{array}$$

Consider an initial distribution

$v = [0.1, 0.2, 0.2, 0.15, 0.05, 0.06, 0.14, 0.1]$

→ The stationary distribution is defined as

$$\pi^* = T \pi^*$$

→ It can be obtained as

for i in range (max_iters)
$\pi_{i+1} = T \pi_i$

check_converge():
break

→ Convergence is when $\pi_{i+1} = \pi_i$ for a few iterations

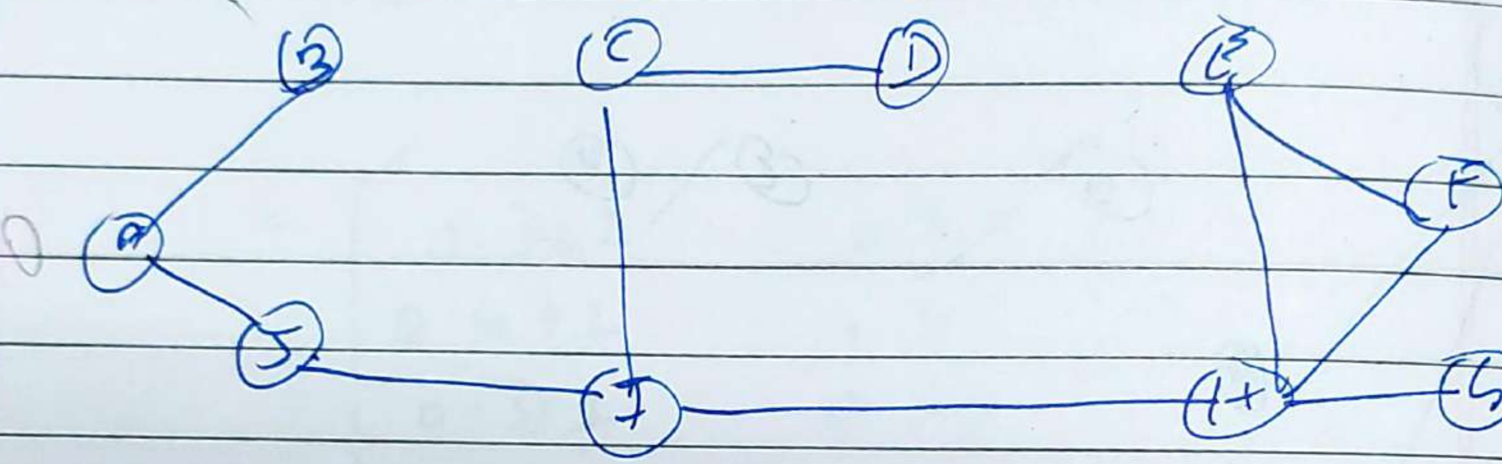
→ The stationary distribution is

[0.111, 0.111, 0.111, 0.1667, 0.1667,
0.111, 0.111, 0.111]

Q2 GCN Graph

18

2)



The embedding of a GCN is given by

$$H^{l+1} = \sigma\left(D^{-1/2}\tilde{A}D^{-1/2}H^{l}W^{l}\right)$$

$$D^{-1/2}AD^{-1/2} = \bar{A} \quad \Big| \quad \tilde{A} = A + I$$

$$H^{l+1} = \sigma\left(\bar{A}H^{l}W^{l}\right)$$

In General a GCN would consist of a MLP and a corresponding Softmax at the ~~final~~ output of ~~a~~ the final layer.

for Simplicity we assume the output of the final layer (i.e the result of the Sigmoid function) to be the embedding as well as the resulting prediction.

Q2 Derivation

19

$$L = -\sum y \log \hat{y} + (1-y)\log(1-\hat{y})$$

$\frac{\partial \hat{y}}{\partial w}$ $\quad \hat{y} = H^{l+1}$

$$H^{l+1} = \sigma(D^{-1/2} A D^{-1/2} H^{l} W^{l})$$

$$H^{l+1} = \sigma(\bar{A} H^{l} W^{l})$$

$$\frac{\partial L}{\partial W^{l}} = -\sum \left(\frac{y}{\hat{y}} + \frac{-(1-y)}{1-\hat{y}}\right)\frac{\partial \hat{y}}{\partial W^{l}}$$

$$= -\sum \left[\frac{y(1-\hat{y}) - \hat{y}(1-y)}{\hat{y}(1-\hat{y})}\right]\frac{\partial \hat{y}}{\partial W^{l}}$$

$$= -\sum \frac{y-\hat{y}}{\hat{y}(1-\hat{y})}\frac{\partial \hat{y}}{\partial W^{l}}$$

→ Solution for $\frac{\partial \hat{y}}{\partial W^{l}} = \frac{\partial H^{l+1}}{\partial W^{l}}$

$$H^{l+1} = \sigma(X)$$

$$\frac{\partial H^{l+1}}{\partial W^{l}} = \sigma(X)(1-\sigma(X))\frac{\partial X}{\partial W^{l}}$$

$$= \sigma(X)(1-\sigma(X))\bar{A}H^{l} \rightarrow (1)$$

Date
Page

→ To obtain gradients across multiple layers we use chain rule. i.e

$$\frac{\partial \hat{y}}{\partial w^{l-1}} = \frac{\partial H^{l+1}}{\partial H^{l}} \cdot \frac{\partial H^{l}}{\partial w^{l-1}}$$

$$\frac{\partial H^{l+1}}{\partial H^{l}} = \sigma(x)\,(1-\sigma(x))^{T}\,\frac{\partial x}{\partial H^{l}} \rightarrow (2)$$

Let $F = H^{l} w^{l}$

$$H^{l^T} F = (H^{l^T} H^{l})\, w^{l}$$

$$\frac{\partial H^{l^T} F}{\partial w^{l}} = \frac{\partial}{\partial w^{l}} (H^{l^T} H^{l})\, w^{l}$$

$$F + H^{l^T}\frac{\partial F}{\partial w^{l}} = 2 H^{l} w^{l} \quad [F = H^{l} w^{l}]$$

$$\Rightarrow H^{l^T}\frac{\partial F}{\partial w^{l}} = F$$

$$\frac{\partial F}{\partial w^{l}} = (H^{l} H^{l^T})^{-1} H^{l} w^{l} \rightarrow (3)$$

$$\frac{\partial H^{l+1}}{\partial H^{l}} = \sigma(x)\,(1-\sigma(x))\,\bar{A}\,(H^{l} H^{l^T})^{-1} H^{l} w^{l} \rightarrow (4)$$

$$\frac{\partial \hat{y}}{\partial w^{l+1}} = \underbrace{\left\{\frac{\partial H^{l+1}}{\partial H^{l}}\right\}}_{\text{Eq. }(4)} \underbrace{\left\{\frac{\partial H^{l}}{\partial w^{l-1}}\right\}}_{(1)}$$

Q2 1 Layer GCN

2)

1 Layer GCN [1 Layer GCN Implementation]

Iteration 0

→ Initialization:-

$$v = \begin{bmatrix} 0.5 \\ 0.25 \\ 0.5 \\ 0.66 \\ 1.5 \\ 0.8 \\ 0.33 \\ 0.75 \\ 0.5 \\ 0.8 \end{bmatrix} \xRightarrow[W^{init} \in R^{1\times2}]{V.W^{init}} \begin{bmatrix} 0.25 & 0.165 \\ 0.125 & 0.085 \\ 0.25 & 0.165 \\ 0.33 & 0.217 \\ 0.75 & 0.475 \\ 0.4 & 0.264 \\ 0.165 & 0.108 \\ 0.375 & 0.24 \\ 0.25 & 0.16 \\ 0.4 & 0.24 \end{bmatrix}$$

$W^{init} = [0.5, 0.33]$

H^0

→ Forward pass

$W^0 = [0.6, 0.45]$

$\sigma(\bar{A} W^0$

$$\sigma(\bar{A} H^0 W^0) = \begin{bmatrix} 0.559 \\ 0.536 \\ 0.5647 \\ 0.559 \\ 0.606 \\ 0.606 \\ 0.544 \\ 0.606 \\ 0.57 \\ 0.56 \end{bmatrix} = H^1$$

Sigmoid

Date ______
Page ______

* ~~Backpropagation~~

~~diff~~

→ Classification / Loss :- (Node Classification)

$$L = -\sum y \log \hat{y} + (1-y)\log(1-\hat{y})$$

$$y = [0, 1, 0, 1, 1, 1, 0, 1, 0, 1]$$

$$L = 6.62 \left(\underline{\hat{y} = H^{(1)}} \right)$$

→ Backpropagation :-

$$\cancel{\frac{\partial L}{\partial W^0} = -\sum \frac{y-\hat{y}}{\hat{y}(1-\hat{y})}\, \sigma(x)(1-\sigma(x))}$$

$$\frac{\partial L}{\partial W^0} = -\sum \frac{y-\hat{y}}{(\hat{y})(1-\hat{y})} \frac{\partial H^1}{\partial W^0} \rightarrow (5)$$

$$\frac{\partial H^1}{\partial W^0} = \sigma(x)\left(1-\sigma(x)\right)^T \tilde{A}\, H^0$$

[From eq. 1].

$$\frac{\partial H^1}{\partial W^0} = \begin{bmatrix} 0.76 & 0.50 \\ 0.73 & 0.48 \\ 0.77 & 0.51 \\ 0.76 & 0.50 \\ 0.824 & 0.54 \\ 0.83 & 0.55 \\ 0.74 & 0.49 \\ 0.827 & 0.54 \\ 0.78 & 0.51 \\ 0.76 & 0.50 \end{bmatrix}$$

Substitute in 5 we get

$$\frac{\partial L}{\partial w^0} = - [1.20, 0.79]$$

$$w^0 = w^0 - \eta \frac{\partial L}{\partial w^0}$$

$$= [0.6, 0.41] + 0.1 [1.2, 0.79]$$

$$= [0.72, \underline{\underline{0.53}}]$$

Iteration 2

→ using the same formulation described previously we get

→ Forward pass

$$\sigma(\tilde{A} H^0 W^0) = \begin{bmatrix} 0.548 \\ 0.529 \\ 0.55 \\ 0.548 \\ 0.586 \\ 0.587 \\ 0.53 \\ 0.58 \\ 0.56 \\ 0.55 \end{bmatrix} = H'$$

24

→ ~~Backpropagation~~ :-

→ Loss = Cross entropy loss = ~~[illegible]~~ $\underline{\underline{6.52}}$

→ Backpropagation :-

$$\frac{\partial H^1}{\partial w^0} = \begin{bmatrix} 0.75 & 0.476 \\ 0.91 & 0.47 \\ 0.75 & 0.50 \\ 0.75 & 0.49 \\ 0.82 & 0.54 \\ 0.83 & 0.543 \\ 0.72 & 0.48 \\ 0.82 & 0.54 \\ 0.77 & 0.51 \\ 0.75 & 0.50 \end{bmatrix}$$

$$\frac{\partial L}{\partial w^0} = -[0.83,\ 0.55]$$

$$w^0 = w^0 - \eta \frac{\partial L}{\partial w^0}$$

$$= [0.72,\ 0.53] + 0.1\,[0.83,\ 0.55]$$

$$= [0.803,\ \underline{\underline{0.585}}]$$

→ Iteration 3

~~$\tilde{A}$~~

Forward pass

$$\sigma(\tilde{A} H^0 W^0) = \begin{bmatrix} 0.579 \\ 0.54 \\ 0.585 \\ 0.57 \\ 0.63 \\ 0.64 \\ 0.58 \\ 0.65 \\ 0.59 \\ 0.58 \end{bmatrix} = H^{l+1}$$

→ Loss : 0.50

→ Backpropagation:-

$$\frac{\partial H^1}{\partial W^0} = \begin{bmatrix} 0.79 & 0.47 \\ 0.70 & 0.46 \\ 0.75 & 0.49 \\ 0.79 & 0.48 \\ 0.81 & 0.54 \\ 0.82 & 0.55 \\ 0.71 & 0.47 \\ 0.84 & 0.54 \\ 0.76 & 0.50 \\ 0.75 & 0.49 \end{bmatrix}$$

26

$$\frac{\partial L}{\partial w^{0}} = -[0.53, 0.38]$$

$$w^{0} = w^{0} - \eta \frac{\partial L}{\partial w^{0}}$$

$$= \cancel{[0.853]}$$

$$[0.803, 0.581] + 0.1 [0.53, 0.38]$$

$$w^{0} = [0.856, 0.623]$$

Q2 2 Layer GCN

(2)

2 Layer GCN

→ Initialization:-

$$V = \begin{bmatrix} 0.5 \\ 0.25 \\ 0.5 \\ 0.66 \\ 1.5 \\ 0.8 \\ 0.33 \\ 0.75 \\ 0.5 \\ 0.8 \end{bmatrix} \quad VW^{init} \Rightarrow \begin{bmatrix} 0.25 & 0.165 \\ 0.125 & 0.08 \\ 0.25 & 0.165 \\ 0.33 & 0.21 \\ 0.75 & 0.49 \\ 0.4 & 0.26 \\ 0.165 & 0.108 \\ 0.375 & 0.24 \\ 0.25 & 0.165 \\ 0.4 & 0.26 \end{bmatrix} = H^0$$

~~winit~~

$W^{init} = [0.5, 0.33]$

→ Iteration 1

Forward pass:-

$$H^1 = \sigma(\tilde{A} H^0 W^0)$$

$$W^0 = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$H^1 = \begin{pmatrix} 0.519 & 0.53 \\ 0.552 & 0.52 \\ 0.526 & 0.533 \\ 0.519 & 0.530 \\ 0.535 & 0.55 \\ 0.54 & 0.55 \\ 0.515 & 0.55 \\ 0.54 & 0.524 \\ 0.52 & 0.539 \\ 0.521 & 0.534 \end{pmatrix}$$

$$H^2 = \cancel{\sigma(\bar{A} H^1 W^1)} \quad \sigma(\bar{A} H^1 W^1)$$

$$W^1 = [\,0.4 \quad 0.5\,]^T$$

$$H^2 = \begin{pmatrix} 0.62 \\ 0.60 \\ 0.62 \\ 0.60 \\ 0.61 \\ 0.61 \\ 0.59 \\ 0.64 \\ 0.62 \\ 0.61 \end{pmatrix} = \hat{y}$$

→ Loss = using cross entropy loss.

$= 6.74$

→ <u>Backpropagation:</u> [update weights for W^1]

$$\frac{\partial H^2}{\partial W^1} = \sigma(x)\,(1-\sigma(x))^T \bar{A} H^1$$

$$\frac{\partial H^2}{\partial W^1} = \begin{bmatrix} 1.23 & 1.26 \\ 1.19 & 1.22 \\ 1.22 & 1.25 \\ 1.19 & 1.23 \\ 1.20 & 1.24 \\ 1.21 & 1.25 \\ 1.17 & 1.20 \\ 1.28 & 1.31 \\ 1.23 & 1.26 \\ 1.20 & 1.24 \end{bmatrix}$$

Substitute in 5 we get

$$\frac{\partial H^2}{\partial W^1} = -[-0.83, -0.85]$$

$$= [0.83, 0.85]$$

$$W^1 = W^1 - \eta \frac{\partial L}{\partial W^1}$$

$$= [0.4, 0.5] - 0.1\,[0.83, 0.85]$$

$$= [0.317, 0.415]$$

[Update weights for W^1]

$$\frac{\partial H^2}{\partial H^1} = \underset{10\times1}{\sigma(x)}\ \underset{1\times10}{(1-\sigma(x))^T}\ \underset{10\times10}{\bar{A}}\ \underset{(10\times10)}{(H^1 H^{1T})^{-1}}\ \underset{(10\times2)}{H^1}\ \underset{(2\times1)}{W^1}$$

$=$

Update weigh for W^0

Extending Equation 1 for matrices we get-

$$\frac{\partial H^{l+1}}{\partial W^l} = \sigma(x)\,(1-\sigma(x))\,\frac{\partial x}{\partial W^l}$$

$$x = \bar{A} H^l W^l$$
$$x W^{lT} = \bar{A} H^l (W^l W^{lT})$$

$$W^{lT}\frac{\partial x}{\partial W^l} + x = 2\bar{A} H^l W^l$$

$$W^{lT}\frac{\partial x}{\partial W^l} W^T = \bar{A} H^l W^l$$

$$\frac{\partial x}{\partial W^l} = (W^{lT} W^l)^{-1}(x\bar{A} H^l W^l)$$

$$\frac{\partial x}{\partial W^l} = (W^l W^{lT})^{-1} W^l \bar{A} H^l W^l$$

Update weight for w°

Extending Equation 1 for matrices we get-

$$\frac{\partial H^{l+1}}{\partial w^{l}} = \sigma(x)\,(1-\sigma(x))\,\frac{\partial x}{\partial w^{l}}$$

$$X = \bar{A} H^{l} w^{l}$$
$$X w^{l^T} = \bar{A} H^{l} (w^{l} w^{l^T})$$

$$\cancel{\frac{\partial H^{l+1}}{\partial w^l} = \sigma(x)(1-\sigma(x))(w^l w^{l^T})^{-1} w^l \bar{A} H^l w^l}$$

$$\frac{\partial x}{\partial w^l} w^{l^T} + x = 2x$$

$$\frac{\partial x}{\partial w^l} w^{l^T} = \bar{A} H^l w^l$$

$$\frac{\partial x}{\partial w^l} = \bar{A} H^l w^l w^l (w^{l^T} w^l)^{-1}$$

$$= \bar{A} H^l w^l \underbrace{w^l . w^{l^{-1}}} . w^{l^{T^{-1}}}$$

$$\left(\frac{\partial x}{\partial w^l}\right)^T = \left(\underline{\bar{A} H^l w^l w^{l^{T^{-1}}}}\right)^T$$

$$\frac{\partial H^{l+1}}{\partial w^l} = \cancel{\sigma(x)(1-\sigma(x))^T \bar{A} H^l w^l} = \underline{(w^l w^{l^{T^{-1}}})} \left(\sigma(x)^T (1-\sigma(x))^T \bar{A} H^l\right)$$

$\in R^{2\times 2}$

$$\frac{\partial L}{\partial w^0} = -\sum \frac{y-\hat{y}}{\hat{y}(1-\hat{y})} \left\{\frac{\partial H}{\partial H^1}\right\}^T \left\{\frac{\partial H^1}{\partial w^0}\right\}$$

$R^{2\times 10}$ $\quad R^{10\times 2}$ $\quad \hookrightarrow R^{2\times 2}$ (6)

$$w^0 = w^0 - \eta \frac{\partial L}{\partial w^0}$$

[Substituting values in the above equation we get]

$$= \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix} - 0.1 \begin{bmatrix} 0.21 & 0.79 \\ 0.05 & 0.21 \end{bmatrix}$$

$$= W^0 = \begin{bmatrix} 0.08 & 0.18 \\ 0.25 & 0.38 \end{bmatrix}.$$

$$W^1 = [0.317, 0.415]$$

[updated weights]

→ Iteration 2 :-

Forward pass :-

$$H^1 = \sigma(AH^0W^0)$$

$$= \begin{bmatrix} 0.514 & 0.53 \\ 0.510 & 0.51 \\ 0.517 & 0.532 \\ 0.516 & 0.534 \\ 0.529 & 0.55 \\ 0.530 & 0.56 \\ 0.512 & 0.53 \\ 0.532 & 0.56 \\ 0.521 & 0.59 \\ 0.517 & 0.59 \end{bmatrix}$$

$$H^2 = \sigma(A H^1 w^1)$$

$$= \begin{bmatrix} 0.6 \\ 0.58 \\ 0.59 \\ 0.58 \\ 0.592 \\ 0.589 \\ 0.57 \\ 0.62 \\ 0.60 \\ 0.59 \end{bmatrix}$$

Loss function = $\underline{\underline{6.73}}$

<u>Backpropagation:-</u>

→ update for weights w^1

$$\frac{\partial H^2}{\partial w^1} = \sigma(x)\,(1-\sigma(x))^T \bar{A} H^1$$

using the procedure explained in previous iteration we get

$$\frac{\partial L}{\partial w^1} = -[-0.91, -0.814]$$

$$\frac{\partial L}{\partial W^0} = \begin{bmatrix} 0.3 & 0.05 \\ 0.2 & -0.03 \end{bmatrix}$$

$$W^0 = W^0 - \eta \frac{\partial L}{\partial W^0}$$

$$= \begin{bmatrix} 0.08 & 0.18 \\ 0.25 & 0.38 \end{bmatrix} - 0.1 \begin{bmatrix} 0.3 & 0.05 \\ 0.2 & -0.03 \end{bmatrix}$$

$$= \begin{bmatrix} 0.05 & 0.175 \\ 0.23 & 0.41 \end{bmatrix}$$

$$W^1 = W^1 - \eta \frac{\partial L}{\partial W^1} = [0.408, \ 0.475]$$

updated weights:-

$$W^0 = \begin{bmatrix} 0.05 & 0.175 \\ 0.23 & 0.41 \end{bmatrix}$$

$$W^1 = [0.408, \ 0.496]$$

Iteration 3

~~Iteration~~

→ Forward pass

$H^1 = \sigma(\bar{A} H^0 W^0)$

$$= \begin{bmatrix} 0.51 & 0.527 \\ 0.508 & 0.516 \\ 0.51 & 0.529 \\ 0.543 & 0.527 \\ 0.524 & 0.548 \\ 0.539 & 0.549 \\ 0.51 & 0.52 \\ 0.52 & 0.54 \\ 0.516 & 0.53 \\ 0.514 & 0.529 \end{bmatrix}$$

$H^2 = \sigma(\bar{A} H^1 W^1)$

$$= \begin{bmatrix} 0.62 \\ 0.60 \\ 0.61 \\ 0.60 \\ 0.62 \\ 0.61 \\ 0.59 \\ \cancel{0.58}\ 0.64 \\ 0.58 \\ 0.62 \\ \cancel{0.64} \end{bmatrix}$$

Date
Page

36

Loss = 6.72

Back propagation :-

update for weight W^0

$\frac{\partial L^2}{\partial w^1}$ = ~~[-0.8]~~
= - [-0.76, -0.84]

update for weight W^1

$$\frac{\partial L}{\partial w^1} = \begin{bmatrix} 0.05 & 0.32 \\ 0.18 & 0.45 \end{bmatrix}$$

$$w^1 = w^1 - \eta \frac{\partial L}{\partial w^1}$$

$$= \begin{bmatrix} 0.045 & 0.143 \\ 0.212 & 0.365 \end{bmatrix}$$

$w^0 = w^0 - \eta \frac{\partial L}{\partial w^0}$ = ~~[0.484]~~

= [0.332, 0.412]

W^1 = [0.332, 0.412]

$$W^0 = \begin{bmatrix} 0.045 & 0.143 \\ 0.212 & 0.365 \end{bmatrix}$$

Q2 3 layer GCN

3 layer GCN

→ Iteration 1

Forward pass

Initialization of weights

$$W^0 = \begin{bmatrix} 0.1 & 0.2 \\ 0.3 & 0.4 \end{bmatrix}$$

$$W^1 = \begin{bmatrix} 0.02 & 0.15 \\ 0.32 & 0.22 \end{bmatrix}$$

$$W^2 = \begin{bmatrix} 0.16, & 0.34 \end{bmatrix}$$

$$H^1 = \sigma(\bar{A} H^0 W^0)$$
$$H^2 = \sigma(\bar{A} H^1 W^1)$$
$$H^3 = \sigma(\bar{A} H^2 W^2)$$

H^0 ~~is~~ is obtained from the procedure explained previously using W^{init}.

$$= \begin{bmatrix} 0.569 \\ 0.55 \\ 0.567 \\ 0.559 \\ 0.563 \\ 0.563 \\ 0.55 \\ 0.58 \\ 0.57 \\ 0.56 \end{bmatrix}$$

Loss = 6.75

→ **Backpropagation:-**

Using the formulation described earlier we get the gradients for every layer as shown below.

$$\frac{\partial L}{\partial W^0} = -\begin{bmatrix} 0.15 & 0.18 \\ 0.14 & -0.06 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^1} = -\begin{bmatrix} 0.21 & 0.24 \\ -0.13 & 0.04 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^2} = -\begin{bmatrix} 0.45 & 0.32 \end{bmatrix}$$

$$W^0 = W^0 - \eta \frac{\partial L}{\partial W^0} = \begin{bmatrix} 0.115 & 0.218 \\ 0.314 & 0.394 \end{bmatrix}$$

$$W^1 = W^1 - \eta \frac{\partial L}{\partial W^0} = \begin{bmatrix} 0.041 & 0.174 \\ 0.307 & 0.216 \end{bmatrix}$$

$$W^2 = W^2 - \eta \frac{\partial L}{\partial W^2} = \begin{bmatrix} 0.205 & 0.374 \end{bmatrix}$$

Loss = 6.75

→ **Backpropagation:-**

Using the formulation described earlier we get the gradients for for every layer as shown below.

$$\frac{\partial L}{\partial W^0} = -\begin{bmatrix} 0.15 & 0.18 \\ 0.14 & -0.06 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^1} = -\begin{bmatrix} 0.21 & 0.24 \\ -0.13 & 0.04 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^2} = -\begin{bmatrix} 0.45 & 0.32 \end{bmatrix}$$

$$W^0 = W^0 - \eta \frac{\partial L}{\partial W^0} = \begin{bmatrix} 0.115 & 0.218 \\ 0.314 & 0.394 \end{bmatrix}$$

$$W^1 = W^1 - \eta \frac{\partial L}{\partial W^0} = \begin{bmatrix} 0.041 & 0.174 \\ 0.307 & 0.216 \end{bmatrix}$$

$$W^2 = W^2 - \eta \frac{\partial L}{\partial W^2} = \begin{bmatrix} 0.205 & 0.374 \end{bmatrix}$$

Date ______
Page ______

Iteration 2

→ **Forward pass**

$$H^3 = \sigma(\bar{A} H^2 W^2)$$
$$H^2 = \sigma(\bar{A} H^1 W^1)$$
$$H^1 = \sigma(\bar{A} H^0 W^0)$$

$$H^3 = \begin{bmatrix} 0.58 \\ 0.56 \\ 0.57 \\ 0.569 \\ 0.55 \\ 0.57 \\ 0.562 \\ 0.59 \\ 0.58 \\ 0.53 \end{bmatrix}$$

→ Loss = 6.65

→ **Backpropagation**

$$\frac{\partial L}{\partial W^0} = -\begin{bmatrix} 0.18 & 0.42 \\ 0.56 & 0.14 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^1} = -\begin{bmatrix} 0.22 & 0.12 \\ 0.41 & 0.15 \end{bmatrix}$$

$$\frac{\partial L}{\partial W^2} = -\begin{bmatrix} 0.15 & 0.62 \end{bmatrix}$$

$$W^0 = \begin{bmatrix} 0.133 & 0.258 \\ 0.373 & 0.408 \end{bmatrix}$$

$$W^1 = \begin{bmatrix} 0.063 & 0.186 \\ 0.348 & 0.231 \end{bmatrix}$$

$$W^2 = \begin{bmatrix} 0.19, & 0.436 \end{bmatrix}$$

Iteration 3

→ Forward pass :-

$\cancel{H^3 = 0}$

$$H^3 = \begin{bmatrix} 0.58 \\ 0.57 \\ 0.58 \\ 0.575 \\ 0.5580 \\ 0.58 \\ 0.568 \\ 0.600 \\ 0.59 \\ 0.58 \end{bmatrix}$$

→ Loss = 6.42

→ Backpropagation :-

$$\frac{\partial L}{\partial w^0} = -\begin{bmatrix} 0.15 & 0.62 \\ 0.11 & 0.42 \end{bmatrix}$$

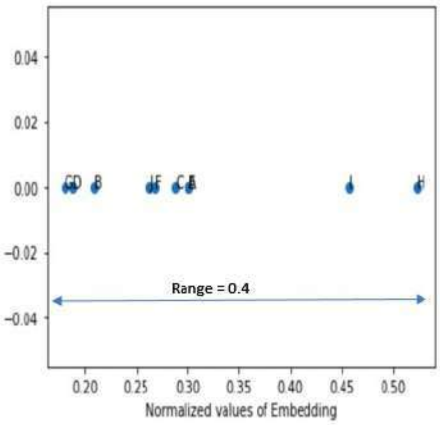
$$\frac{\partial L}{\partial w^1} = -\begin{bmatrix} 0.41 & 0.32 \\ 0.39 & 0.44 \end{bmatrix}$$

$$\frac{\partial L}{\partial w^2} = -\begin{bmatrix} 0.1 & 0.04 \end{bmatrix}$$

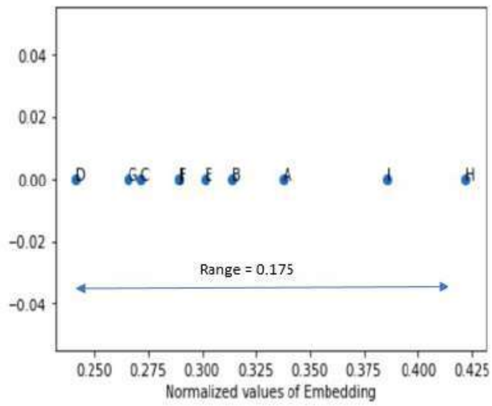
$$w^0 = \begin{bmatrix} 0.148 & 0.32 \\ 0.384 & 0.45 \end{bmatrix}$$

$$w^1 = \begin{bmatrix} 0.104 & 0.218 \\ 0.437 & 0.275 \end{bmatrix}$$

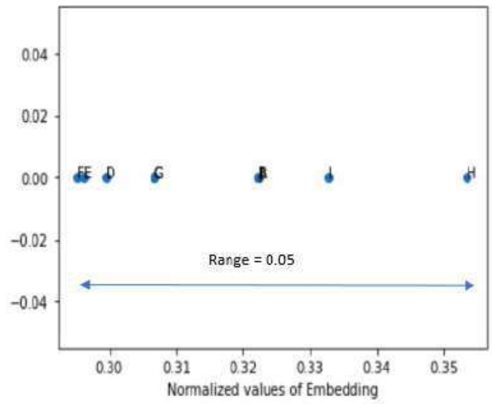
$$w^2 = \begin{bmatrix} 0.2 & 0.44 \end{bmatrix}$$



GCN 1 Layer



GCN 2 Layer



GCN 3 Layer

Q 2.2

42

2.2) → For the given graph the best ~~suited~~ model is the 1 Layer GCN, followed by 2 Layer GCN.

→ The 3 Layer GCN is least suitable.

→ Reasons:-

* The 1 Layer GCN ~~not~~ is able to train and predict lables and maintain ~~uni~~ unique representation for each node.

* We can observe from the above figure ~~(Fig)~~ that the range of Embedding values decrease with increase in ~~Embedde~~ number of layers.

* The ~~Smoothi~~ Smoothning problem seems to have started with 2 Layers, but the two layer model is ~~not~~ still able to maintain accuracy and unique representation.

* The 3 layer ~~not~~ model has a very low ~~val~~ range and ~~it~~ is a clear sign of the smoothning problem.

Q. 2.3

43

* Thus 1 Layer is the best suited over the others as it has least effect of smoothing.

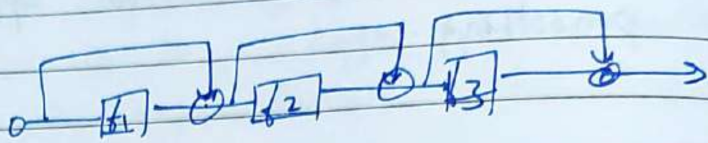
2.3) * With the increase in the ~~to~~ layers of the GCN the receptive field of the network increases

* With the increase in receptive field all the nodes ~~*~~ tend to receive ~~the same~~ ~~fat~~ almost the same messages and thus the same output and same embedding for every node.

* In the given ~~case~~ graph, from ~~fig~~ figure we observe that A 1 layer GCN gives optimal performance, followed by 2 layer (smothning problem starts with 2).

3 ~~suff~~ layers ~~suffers from~~ has the smoothing problem.

* Skip connections



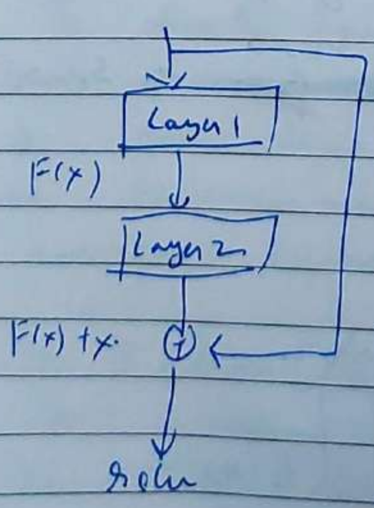
→ Skip connections create a mixture of models

→ N skip connection will have upto 2^N possible paths

→ ~~Ex~~ ~~ea~~ with more and better message passing the GCN will learn to Generate unique embeddings

$$h_v^{(l)} = \sigma\left(\sum_{u \in N(v)} w^l \frac{h_u^{l-1}}{|N(v)|} + h_v^{l-1}\right)$$

Skip connection



-> For the above reasons the skip connections would help ~~to~~ the given graph.

-> The optimal receptive field is with a 1 ~~Layer~~ Layer GCN.

~~-> The given graph has less number of edges, thus the incoming messages~~

-> For the given graph adding skip connections would allow it to get messages from previous layer directly, thus it would help in generating unique embeddings.

-> Furthermore the graph has identical nodes such as A, J (degree 2), connected to non identical neighbours such as ~~e.g.~~ B and I (degree 1 and 3 respectively).
A skip connection would help to recieve data directly from far ~~nodes~~ ~~to~~ nodes and this would be useful ~~to~~ for unique embeddings.

Q.3

46

3) 1) Run Question3.py to replicate ~~result~~ results.

For further instructions about how to run the code please refer to README.

2) Row normalization = $A_{row} = D^{-1}A$

coloumn nomlization = $A_{col} = AD^{-1}$

Symmetric normalization = $A_{sym} = D^{-1/2} A D^{-1/2}$

(Normalization Results).

-> During experiments the accuracy of all the three ~~methods~~ methods was comparable with Symmetric normalization being the highest at 70.1%, followed by row normalization at 69.5% and collumn normalization at 69.1%.

-> A_{sym} gives does some form of average, the square root ensures eigen value = 1.

-> The normalization ensures the values are not very large and can help to improve accuracy.

3)

→ It can be observed accuracy of GCN is ~~this~~ higher in comparison to GraphSage.

→ The main reason could be that GCN is more effective in the combination step. ~~ie mean & sum~~

i.e

mean or sum form of ~~a~~ combination is more effective ~~that~~ than concatination for the Citeseer dataset.

→ With the increase in GCN layers (> 3) ~~Smooth~~ over smoothning problem occurred.

→ ~~The~~ Minor changes in the learning rate ~~[illegible]~~ did not effect the resulting accuracy by much.

4) Please refer to README

5) Please refer to README