USER USER Collaborative filtering

movies →

users ↑	HP1	HP2	HP3	TW	SW1	2	3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Collaborative filtering:

A hasn't watched All the movies

eg. in this case.

A, B ⇒ similar Ratings

A + C : ⇒ conot dissimilar cases

Jaccard Similarity.

$$sim(A,B) = \frac{r_A \cap r_B}{r_A \cup r_B} = \frac{1}{5}$$

$$sim(A,C) = \frac{2}{4}$$

∴ sim (A, B) < Sim(A,C)
⇓
just sees A, B have watched lesser Movies than A,C have in Common & not if they both liked it.
problem

Cosine Similarity

> Insert unknown values : 0.

$$sim(A,B) = \frac{\sum A \cdot B}{\sqrt{A^2}\sqrt{B^2}} = \frac{4*5}{\sqrt{4^2+5^2+1^2} \cdot \sqrt{5^2+5^2+4^2}} = 0.38.$$

sim(A, C) = 0.32 & cosine ∠ b/w both.

∴ sim (A, B) > sim (A, C) but not by much.

problem: treats missing Ratings as least

Centered Cosine / Pearson Correlation / Ratingⁿ Normalizⁿ: Normalization:

A : (4+5+1) / 3 = 10/3

B : (5+5+4) / 3 = 14/3

C : (2+4+5) / 3 = 11/3

D : (3+3) /2 = 6/2

Subtract this value from each cell.

i.e. $4 - \frac{10}{3} = \frac{2}{3}$, $5 - \frac{10}{3} =$ - - -

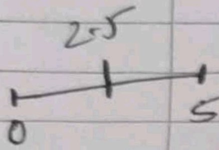
low Rating becomes -ve
High -ve +ve
then Subtract from Avg

Can use Euclidean dist as well

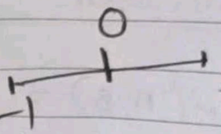
	HP1	Hp2	HP3	TW	SW1	SW2	SW3	
A	2/3			5/3	-7/3			→ $\Sigma = 0$
B	1/3	1/3	-2/3					$\Sigma = 0$
C				-5/3	1/3	4/3		✓
D		0					0	✓
								✓

this makes Avg. Rating ∀ users = 0.

∴



Before	After
1) Avg = 2.5	1) Avg = 0.
2) Range = 0-5	2) -1 to +1
3) 0-2.5 = Below Avg	3) Below 0 : Below Avg.
4) Missing = 0	4) Missing = Avg.



Now Compute Cosines.

$$Sim(A,B) = \frac{\frac{2}{3} * \frac{1}{3}}{\sqrt{\left(\frac{2}{3}\right)^2 + \frac{5}{3})^2 + \left(\frac{7}{3}\right)^2} \cdot \sqrt{\left(\frac{1}{3}\right)^2 + \left(\frac{1}{3}\right)^2 + \left(\frac{-2}{3}\right)^2}}$$

$$= \frac{2/9}{140/729} \simeq 0.092$$

$Sim(A,C) = -0.56$.

$\therefore$ Sim(A,B) > Sim(A,C).

ITEM ITEM Collaborative filtering

(More useful, outperforms user-user CF)

∵ items are simpler than users

- items : fixed set of genres.
- users : kinda like fuzzy logic Nature.

Because you watched Crime patrol, you might also like Savdhaan India.

precision @ k , Recall @ k

$\frac{\#k \text{ Recomm.}}{\#\text{Actual Relevant}}$

estimate Rating done by users5 for movie1

=3.6
3.16
3
.4
3
.6

movies	Users 1	2	3	4	5	6	7	8	9	10	11	12
1	1		3		?	5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

	1	2	3	4	⇓ 5	6	7	8	9	10	11	12
1	-2.6	0	-0.6	0	?	1.4	0	0	1.4	0	0.4	0
2	0	0	1.84	0.84	0	0.84	0	0	0	-1.16	-2.16	-0.16
3	-1	1	0.	-2	-1	0	0	0	1	0	2	0
4	0	-1.4	0.6	0	1.6	0	0	0.6	0	0	-1.4	0
5	0	0	0.7	-0.3	0.7	-1.3	0	0	0	0	-1.3	1.7
6	-1.6	0	0.4	0	0.4	0	0	-0.6	0	0	1.4	0

sim(M1, other Movies Rated by user 5) =? ∴ 2 movies highest similarity
⇓ (1,5) = ?

sim(m1, M3) = $\frac{2.6*1 + 1.4*1 + 0.4*3}{\sqrt{-u-}\ \sqrt{-u}}$ = 0.41 ✓
(1,3)

↑ take weigted Avgs using these

sim(m1, m4) = - 0.10

sim(m1, m5) = - 0.31

sim(m1, m6) = ——— = 0.59 ✓
(1, 6).

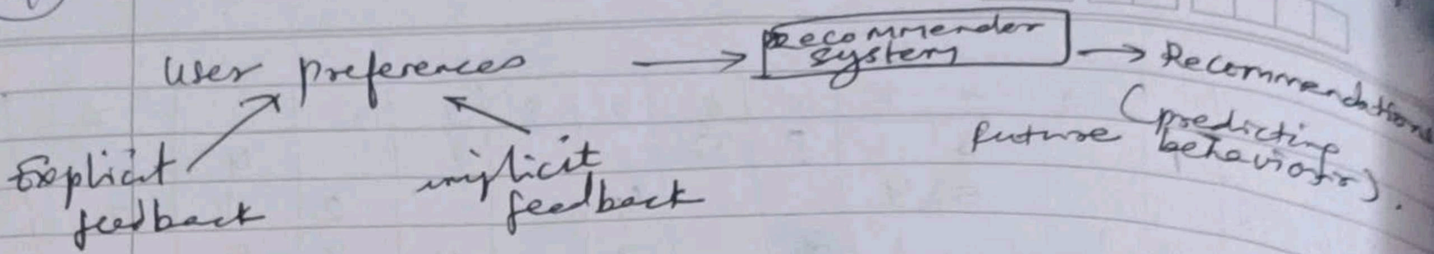
sim(m1, m1) = 1

sim(m1, m2) = - 0.18 .

∴ Ratings for $\boxed{?}$ ⇒ $\frac{0.41 * \overset{(3,5)}{2} + 0.59 * \overset{(6,5)}{3}}{0.41 + 0.59}$ = 2.6 .

∴ $\boxed{?}$ = 2.6 ;

Theory.



<u>Collaborative :</u>

Similar users like Similar things.
user x item Matrix

<u>Content Based :</u>

Considers items/users features

item x item
movie genre,
Year of Release,
Cast,
director,
prod? house.

user x user
age, gender,
Spoken language

---

1. Content Based filtering.

User vs <u>Feature</u> Matrix * <u>Feature</u> vs Movie
Comedy, Action — Comedy, Action
↓
User vs Movie.

Features come directly.

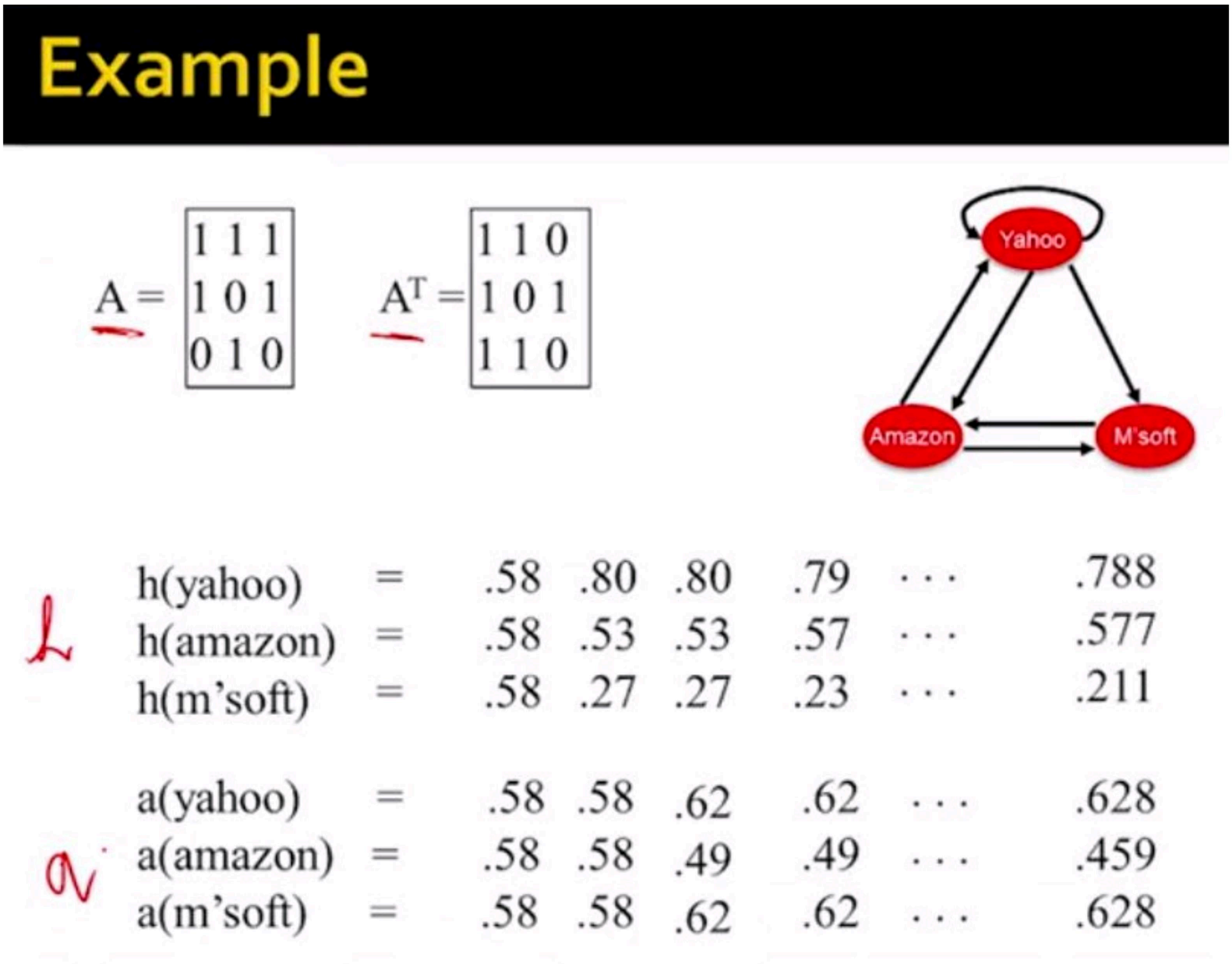
2. Collaborative Based filtering.

User vs Movie with ?
↓ ~~patterns~~ (latent features).
user vs feature * Feature vs Movie
↓
User vs Movie [w/o ? / to fill ?]

you keep guessing values for both Matrixs until you get closest Matrix Multipl? product same as from where u started.

Applications of Recommend? systems : News/Songs/...
eg. Synthetic Control. What'll be effect of "Gun Control" policies if Implemented? you check for countries → to you that already Implemented em

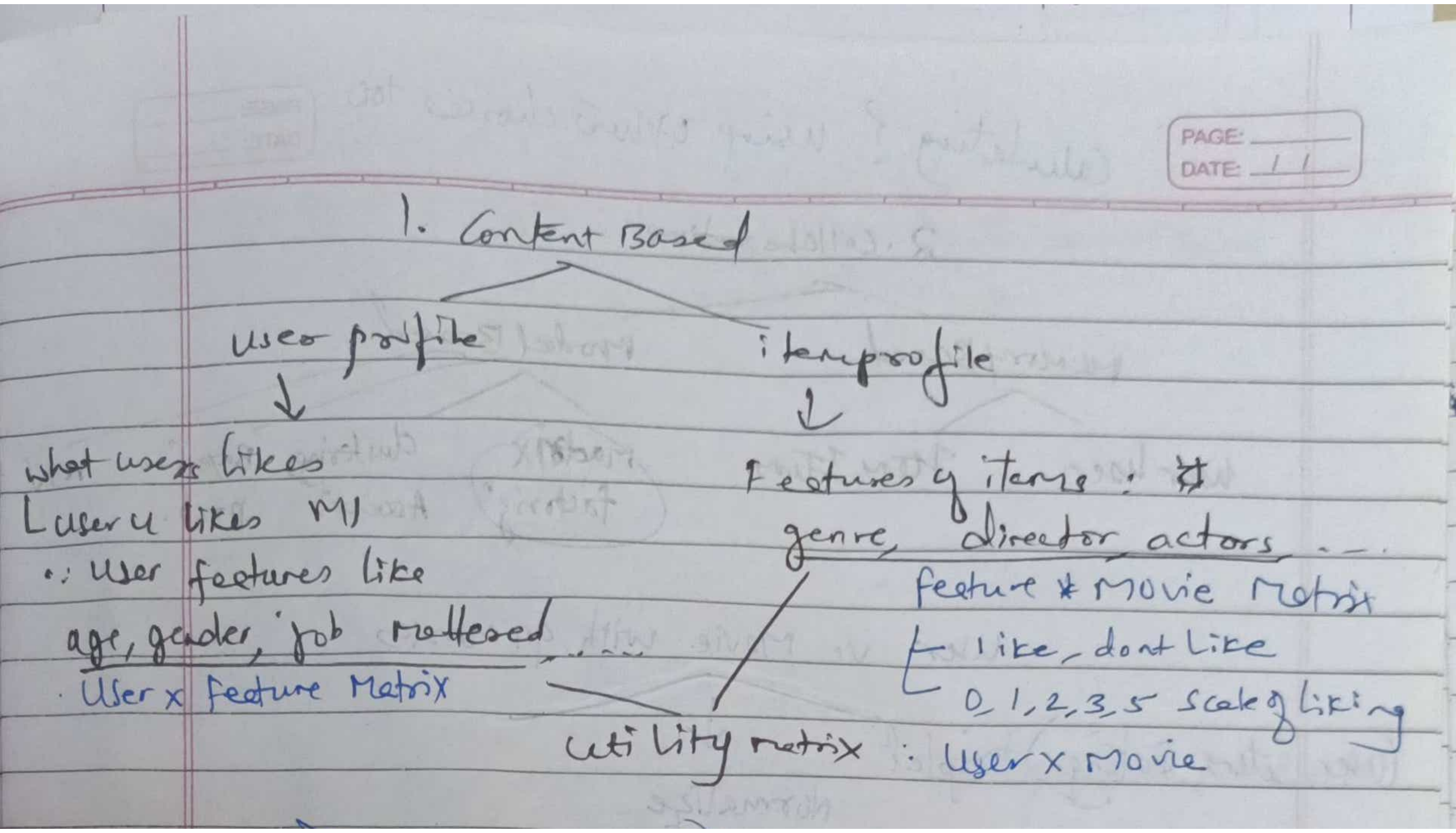
8. Hyperlink Induced Topic Search (HITS) Algorithm



Example

$$\underline{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad \underline{A^T} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

h	h(yahoo)	=	.58	.80	.80	.79	...	.788
	h(amazon)	=	.58	.53	.53	.57	...	.577
	h(m'soft)	=	.58	.27	.27	.23	...	.211
a	a(yahoo)	=	.58	.58	.62	.62	...	.628
	a(amazon)	=	.58	.58	.49	.49	...	.459
	a(m'soft)	=	.58	.58	.62	.62	...	.628



PAGE:
DATE: / /

1. Content Based

user profile

↓

what users likes

L user u likes M1

∴ User features like

age, gender, job mattered, ...

· User x Feature Matrix

itemprofile

↓

Features of items : #

genre, director, actors ...

feature * Movie Matrix

like, dont Like

0, 1, 2, 3, 5 scale of liking

utility matrix ∴ User x Movie

User prefers a good camera more than the phones RAM

user x features * weight

user	likes_Features	w weight of
1	1	2
	3	1
	4	1
2		
⋮		

Oppo phone has a great camera.

product x features * weight.

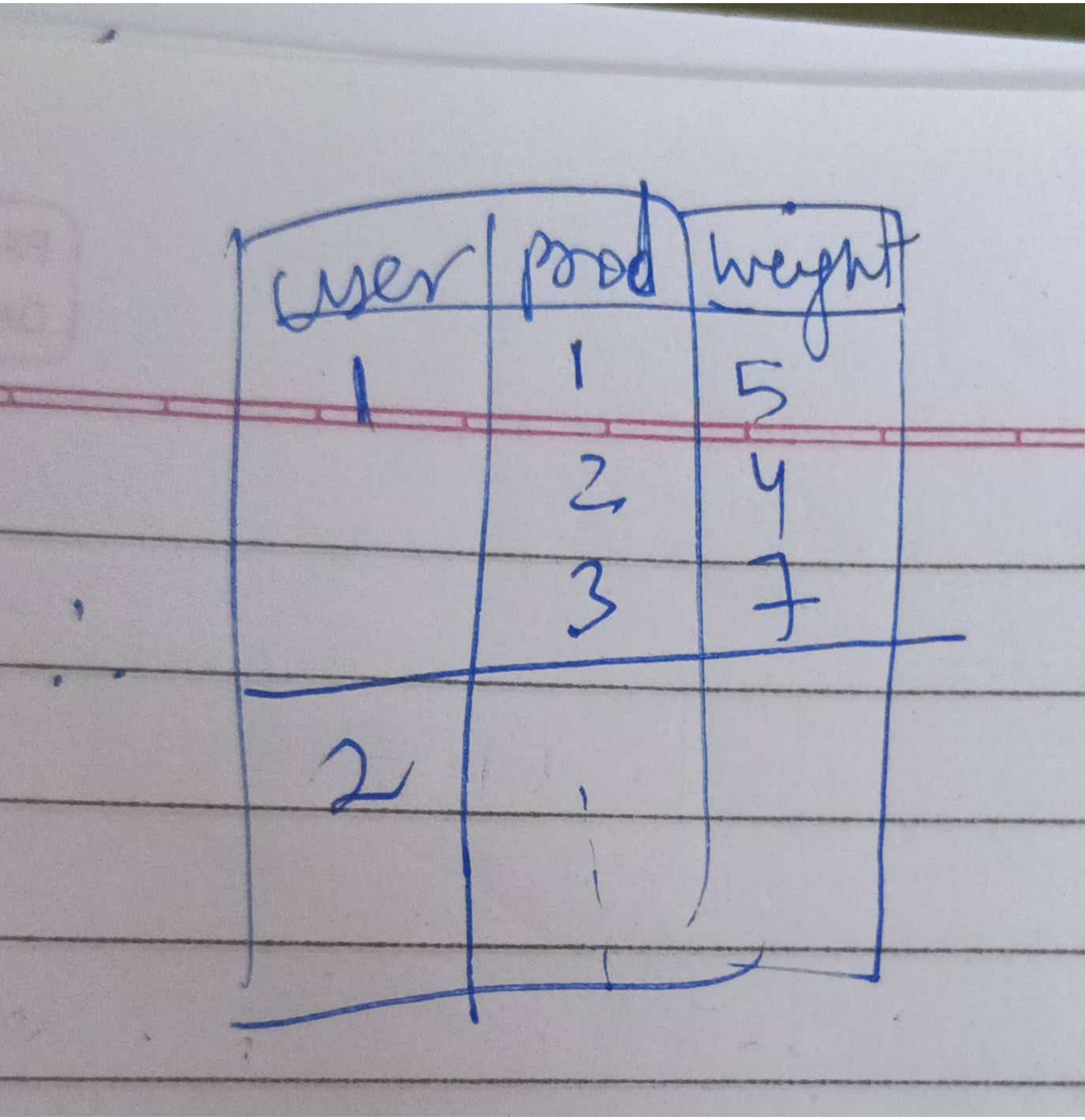
prod1	has_features	order of quality
1	1	1
	3	1
	4	1
2	2	1
	3	4
3	1	3
	4	1

	feature1	feature2	feature3	F4
product1	1		1	1
product2		1	4	
product3	3			1
user1	2		1	1

∴ user Interest in product1 = $2*1 + 1*1 + 1*1 = 5$

$2 = 1*4 = 4$

$3 = 2\times3 + (1) = 7$



user	prod	weight
1	1	5
	2	4
	3	7
2	⋮	

$$\therefore (1,3) = \frac{(1,2) * (2,3) + (1,4) * (4,3)}{(1,2) + (1,4)}$$

collaborative filtering mei...

if sparsity too high, u cant figure out the relations

User \ Movies	1	2	3
1	4	3	2
2	1	3	3
3	5	6	5

Ratings ⇒ Dependent Rows & Cols

1) Pref. of person 3 = Pref of person 1 + 2 (Comedy, Action)

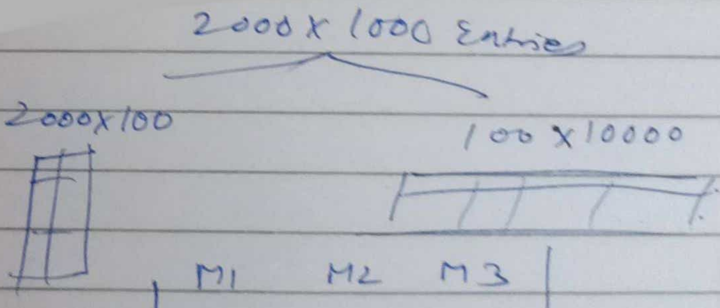
or it possible that $\frac{M1 + M3}{2} = M2$

u try to approximate the original u×i matrix

from u×f and m×f matrix

that were randomly created to fill the ?

from the original u×i matrix



random u×f matrix

random f×m matrix

The Features arent Comedy/Action Explicit?

	f1	F2
u1	0.2	0.5
u2	0.3	0.4
u3	0.7	0.8

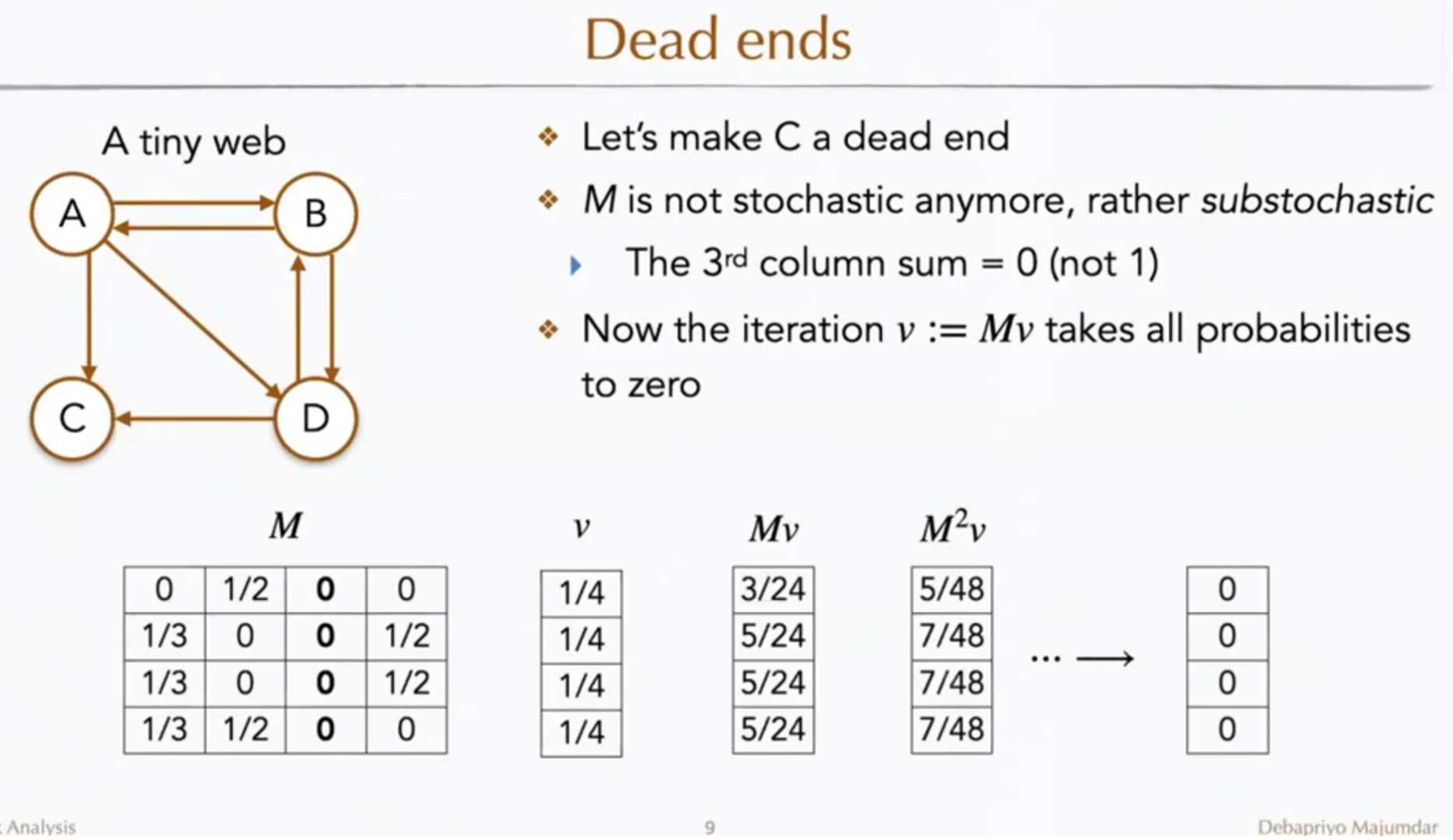
3x2

	M1	M2	M3
f1	1.2	3.1	0.3
F2	2.4	1.5	4.4.

2x3

	M1	M2	M3
u1	.		
u2			
u3			

$P(u_i,f_i)$, (f_i,M_j) to $P(u_i,M_j)$



Dead ends

A tiny web

- Let's make C a dead end
- *M* is not stochastic anymore, rather *substochastic*
  - The 3rd column sum = 0 (not 1)
- Now the iteration $v := Mv$ takes all probabilities to zero

M

0	1/2	**0**	0
1/3	0	**0**	1/2
1/3	0	**0**	1/2
1/3	1/2	**0**	0

v

1/4
1/4
1/4
1/4

Mv

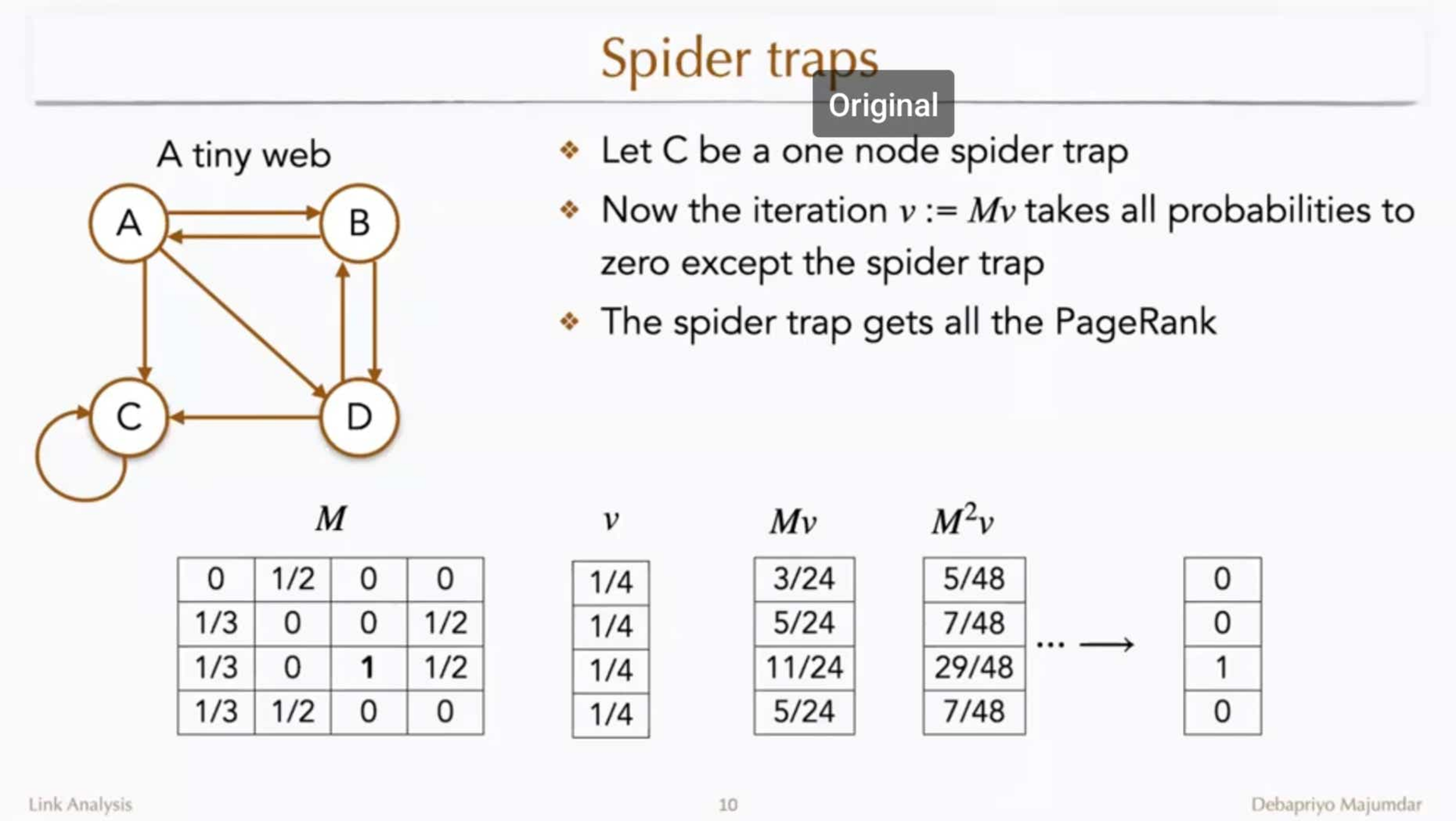
3/24
5/24
5/24
5/24

M^2v

5/48
7/48
7/48
7/48

$\cdots \longrightarrow$

0
0
0
0

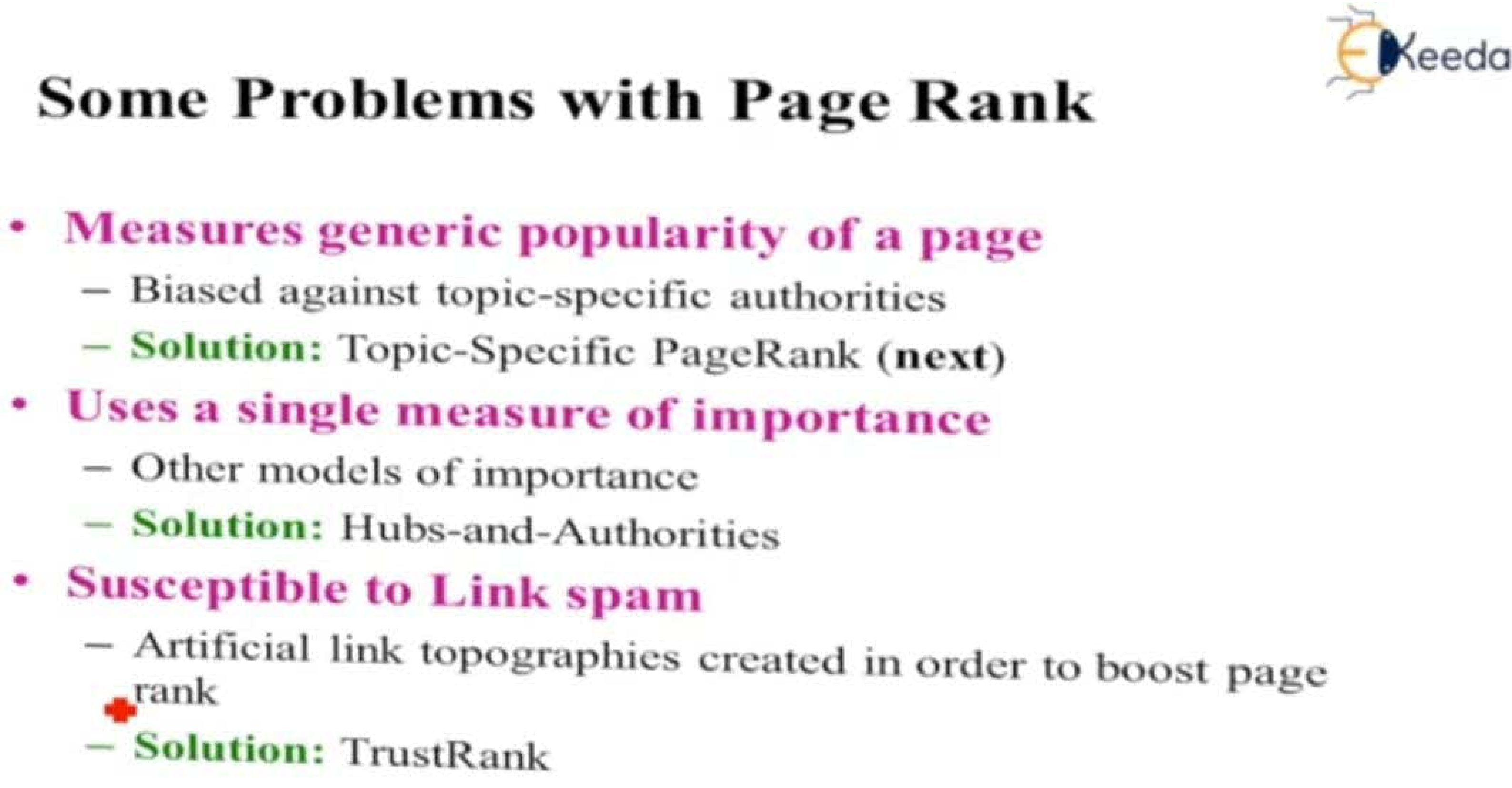


Spider traps

A tiny web

- Let C be a one node spider trap
- Now the iteration $v := Mv$ takes all probabilities to zero except the spider trap
- The spider trap gets all the PageRank

$$M = \begin{array}{|c|c|c|c|} \hline 0 & 1/2 & 0 & 0 \\ \hline 1/3 & 0 & 0 & 1/2 \\ \hline 1/3 & 0 & \mathbf{1} & 1/2 \\ \hline 1/3 & 1/2 & 0 & 0 \\ \hline \end{array} \qquad v = \begin{array}{|c|} \hline 1/4 \\ \hline 1/4 \\ \hline 1/4 \\ \hline 1/4 \\ \hline \end{array} \qquad Mv = \begin{array}{|c|} \hline 3/24 \\ \hline 5/24 \\ \hline 11/24 \\ \hline 5/24 \\ \hline \end{array} \qquad M^2v = \begin{array}{|c|} \hline 5/48 \\ \hline 7/48 \\ \hline 29/48 \\ \hline 7/48 \\ \hline \end{array} \cdots \longrightarrow \begin{array}{|c|} \hline 0 \\ \hline 0 \\ \hline 1 \\ \hline 0 \\ \hline \end{array}$$



Some Problems with Page Rank

- **Measures generic popularity of a page**
  - Biased against topic-specific authorities
  - **Solution:** Topic-Specific PageRank (**next**)
- **Uses a single measure of importance**
  - Other models of importance
  - **Solution:** Hubs-and-Authorities
- **Susceptible to Link spam**
  - Artificial link topographies created in order to boost page rank
  - **Solution:** TrustRank

In this section we shall discuss about the following distance measures in details :

(1) Euclidean Distance (2) Jaccard Distance

(3) Cosine Distance (4) Edit Distance

(5) Hamming Distance

6.1.1 Euclidean Distances

Q. What do you mean by Euclidean distance ? Explain with example.

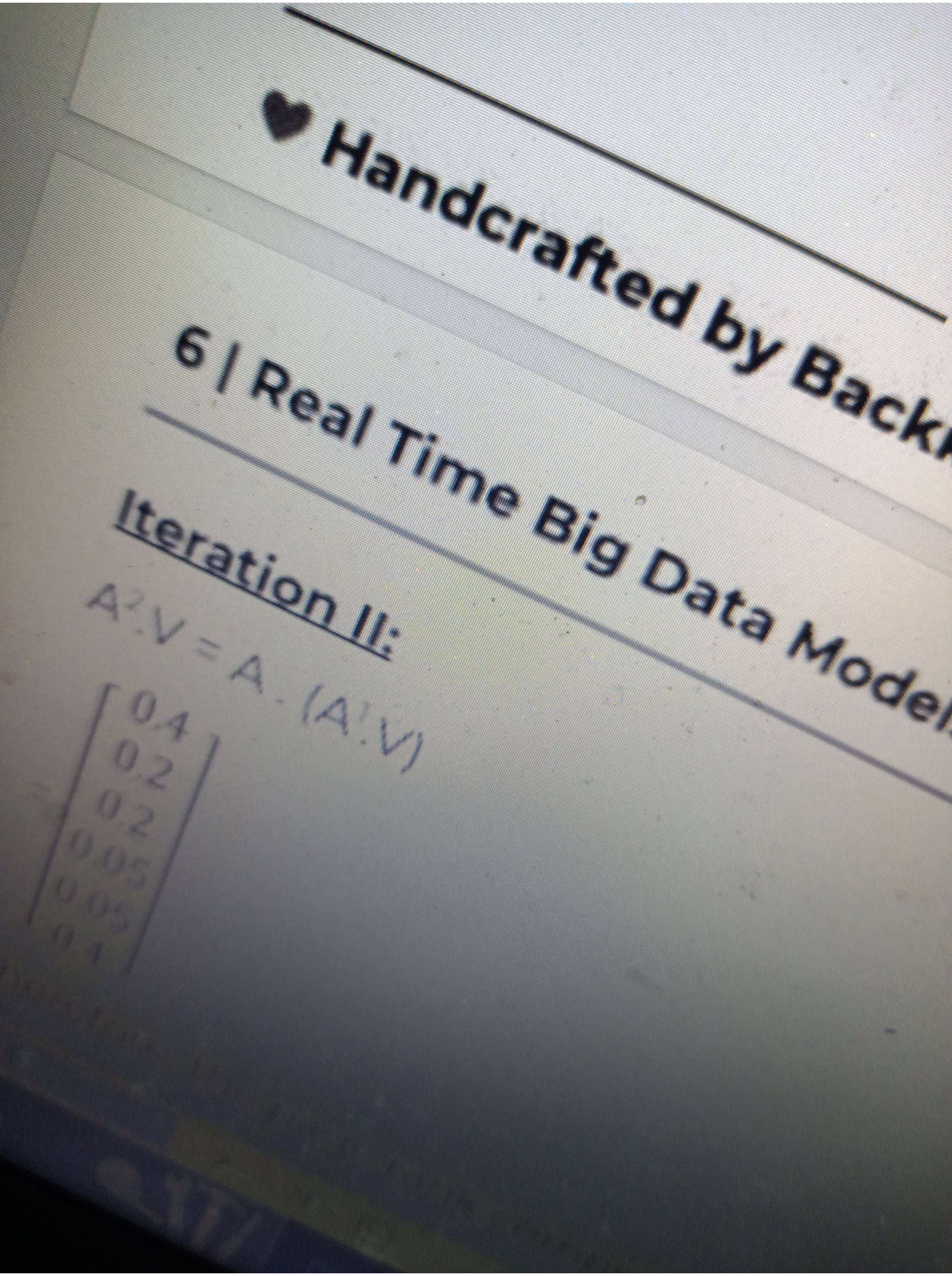
- The Euclidean distance is the most popular out of all the different distance measures
- The Euclidean distance is measured on the Euclidean space. If we consider an n-dimensio then each point in that space is a vector of n real numbers. For example, if we consider t Euclidean space then each point in the space is represented by (x_1, x_2) where x_1 and x_2 are
- The most familiar Euclidean distance measure is known as the L_2– norm which in the n- defined as :

$$d([x_1, x_2, \ldots x_n], [y_1, y_2, \ldots y_n]) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

For the two-dimensional space the L_2– norm will be :

$$d([x_1, x_2], [y_1, y_2]) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

- We can easily verify all the distance axioms on the Euclidean distance :



Iteration II:

$A^2.V = A.(A^1.V)$

$$\begin{bmatrix} 0.4 \\ 0.2 \\ 0.2 \\ 0.05 \\ 0.05 \\ 0.1 \end{bmatrix}$$