

OK PASS

89/69



M T W T F
Page No.:
Date: YOUNA

- If recall increases, precision also increases
- If precision " , recall "

Eg.

		$P(\cdot)$	$R(\cdot)$
+	1	100%	25
+	2	100%	50
-	3	66%	50
+	4	75%	75
-	5	60%	75

Ground truth = 4

- If precision is increasing or staying same, recall increases.

Q Does stemming increase recall or decrease it?

without stemming $|V| = \{ \text{Jump, go, today, jumped, went, jumping} \}$
 $|V| = \{ \text{jump, go, today, went} \}$

Query

$|Q| = \langle \text{Jumping} \rangle$

D1 : Today went jumping

D2 : Go Jump today

D3 : Jumped today

Alphabetical $|V| = \{ \text{go, jump, today, want} \}$

$B_1 = \langle 0, 1, 1, 1 \rangle$

$B_2 = \langle 1, 1, 1, 0 \rangle$

$B_3 = \langle 0, 1, 1, 0 \rangle$

$Q = \langle 0, 1, 0 \rangle$

~~B = (-1, 1, 1)~~

~~$B = \sqrt{2} \hat{i} + \sqrt{3} \hat{j} + \sqrt{3} \hat{k}$~~

$$\cos(B_1, Q) = \frac{1}{\sqrt{3}}$$

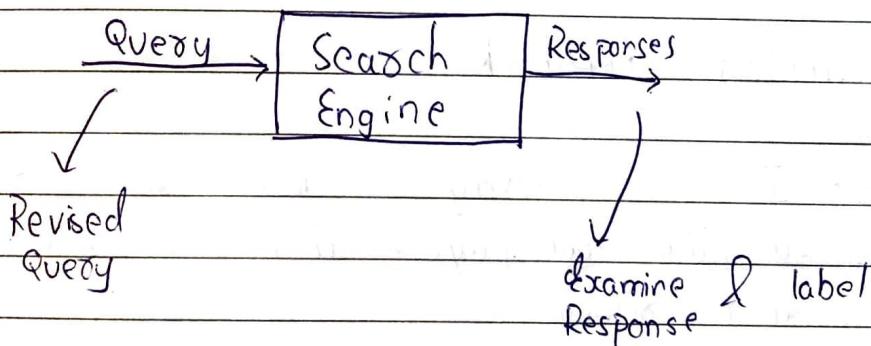
$$\cos(B_2, Q) = \frac{1}{\sqrt{3}}$$

$$\cos(B_3, Q) = \frac{1}{\sqrt{2}}$$



⇒ Relevance feedback.

- Labelled learning



→ We ask user to label the responses, & then we will use labelled / supervised learning.

→ Revised query means for eg. if ~~5~~ ⁵ our query k_1, k_2, \dots, k_5 & we found ~~k3~~ ^{k3} is found in most non-relevant documents, so we ~~will~~ ^{will} revise our query by removing k_3 .

→ If some keywords are not in query but are found most-relevant in web pages then we should revise our query by adding those keyword

⇒ Methods

- 1) Rocchio method (<sup>Mentioned above
adding or removing words</sup>)
- 2) Semi-supervised learning (^{Also known as ~~L-U~~ L-U learning})

2.1) L-U learning

- 2.1) L-U learning → Positive - Unlabelled
- 2.2) P-U learning

→ If user does not label the ~~a~~ web-page then we will ~~use~~ train the model with labelled web-page & will label unlabelled web-page.

3) Implicit Feedback

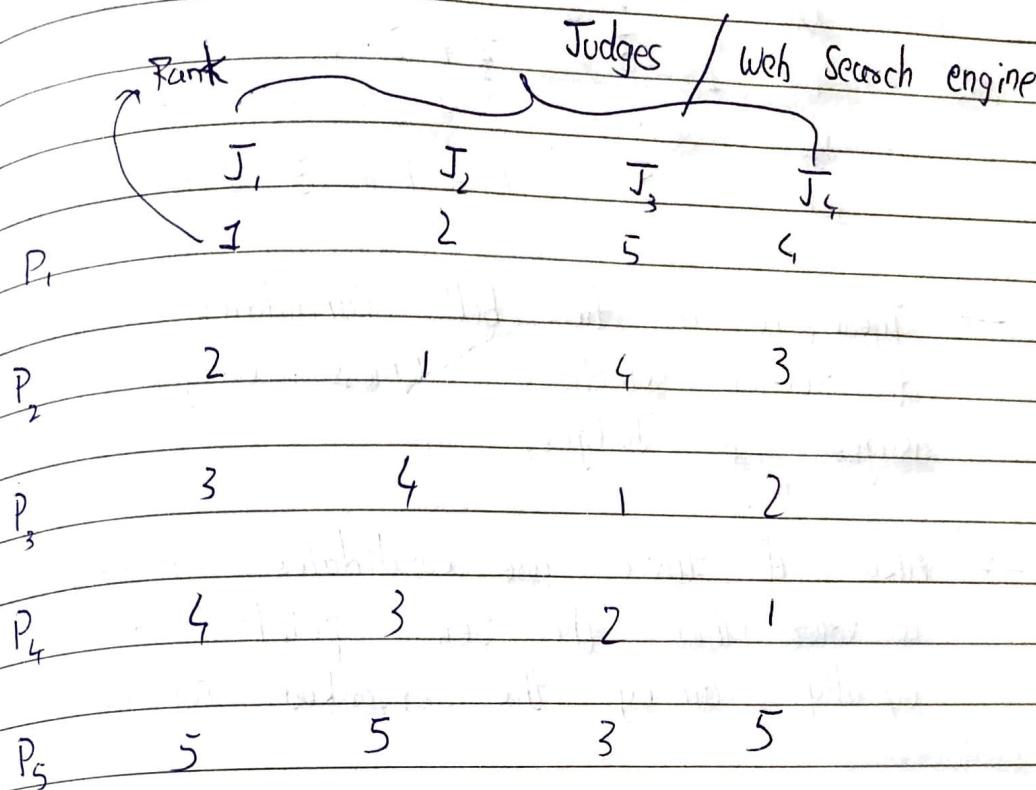
→ If user activity is found ~~is~~ more active in the web-page, then we will ~~no~~ label them relevant.

4) Pseudo - Relevance feedback.

→ In this method we will extract keywords irrespective of whether relevant or not.

→ In this ~~web-pages~~ we are not asking user to provide relevant or N-R document

Meta Search



3 methods

- 1) Borda (1770)
- 2) Reciprocal
- 3) Condorcet

1) Borda

Candidate - Voter method

↓
Participants → Judges.

In the above example we will give n points
acc. to rank for eg.

Rank	Points
1	5
2	4
3	3
4	2
5	1

$$\therefore \text{Candidate 1} = 5 + 4 + 1 + 2 = 12 \text{ points}$$

$$\text{ka " } 2 = 4 + 5 + 2 + 3 = 14 \text{ "}$$

$$\text{" } 3 = 3 + 2 + 5 - 4 = 14 \text{ "}$$

$$\text{" } 4 = 2 + 3 + 4 + 5 = 14 \text{ "}$$

$$\text{" } 5 = 1 + 1 + 3 + 1 = 6 \text{ "}$$

→ There is a tie betⁿ candidates, we will do random selection, unless weights are not assigned to Judges.

→ Also if there are candidates unranked by a voter then the rem. points are divided equally among the unranked candidates.

Eg.

	J ₁	J ₂	J ₃	J ₄
P ₁	1	2	-	4
P ₂	2	1	-	3
P ₃	3	4	1	2
P ₄	4	3	2	1
P ₅	5	5	3	5

Remaining points for J₃ = 2 + 1

↓ Rank 4 ↓ Rank 5

$$= 3$$

∴ Divided Equally ∴ $\frac{3}{2}$

$$\therefore \text{Card 1} = 5 + 4 + \left(\frac{3}{2} \right) + 2$$

$$\text{Card 2} = 4 + 5 + \left(\frac{3}{2} \right) + 3$$

$$\text{Card 3} = 3 + 2 + \left(\frac{3}{2} \right) + 4$$

Some

2) Reciprocal

→ Instead of giving n points to highest rank we give 1 point in reciprocal method

Borda Points distribution :- $n, n-1, n-2, \dots, 1$
 Reciprocal " " :- $1, \frac{1}{2}, \frac{1}{3}, \dots, \frac{1}{n}$

→ For same example.

$$\text{Cand 1} = 1 + \frac{1}{2} + \frac{1}{5} + \frac{1}{4} = 1.95$$

$$\text{Cand 2} = \frac{1}{2} + 1 + \frac{1}{4} + \frac{1}{3} = 2.08$$

$$\text{Cand 3} = \frac{1}{3} + \frac{1}{4} + 1 + \frac{1}{2} = 2.08$$

$$\text{Cand 4} = \frac{1}{4} + \frac{1}{3} + \frac{1}{2} + 1 = 2.08$$

$$\text{Cand 5} = \frac{1}{5} + \frac{1}{4} + \frac{1}{3} + \frac{1}{2} = 0.93$$

→ Here unlike Borda, we will give unranked webpages the value 0.

→ Here also random selection in tie.

→ BOR

Q System 1 :- a, b, c, d

System 2 :- b, a, d, c

System 3 :- c, b, a, d , N=4

System 4 :- c, b, d

System 5 :- c, b

BORDA

$$a = 4 + 3 + 2 + 1 + 1.5 = 11.5 \quad \checkmark$$

$$b = 3 + 4 + 3 + 3 + 3 = 16 \quad \checkmark$$

$$c = 2 + 1 + 4 + 4 + 4 = 15 \quad \checkmark$$

$$d = 1 + 2 + 1 + 2 + 1.5 = 7.5 \quad \checkmark$$

For system 5 , a point

Final Ranking :- B, C, A, D

RECIPROCA)

$$a = 4 + \frac{1}{2} + \frac{1}{3} + 0 + 0 = 1.83$$

$$b = \frac{1}{2} + 1 + \frac{1}{2} + \frac{1}{2} + \frac{1}{2} = 3$$

$$c = \frac{1}{3} + \frac{1}{4} + 1 + 1 + 1 = 3.58$$

$$d = \frac{1}{4} + \frac{1}{3} + \frac{1}{4} + \frac{1}{3} = 1.16$$

Final Ranking :- C, B, A, D

3) Condorcet method.

Step 1 :- Yield an $N \times N$ matrix for all pair wise comparisons

Step 2 :- Calculate the number of wins, losses, tie from each non diagonal entry of the matrix

Step 3:- Determine pairwise winners.

For the same question

Q

Soln

	Pairwise Pairs	S1	S2	S3	S4	S5
out of a & b who has got highest rank for respective systems	a,b	a	b	b	b	b
	a,c	a	a	c	c	c
	a,d	a	a	a	d	-
Building NxN matrix	b,c	b	b	c	c	c
	b,d	b	b	b	b	b
	c,d	c	d	c	c	c

a b c d

a - 1:4:0 2:3:0 3:1:1

Win a

win b

Tie

b - 4:1:0 2:3:0 5:0:0

c - 3:2:0 3:2:0 4:1:0

d - 3:3:1 0:5:0 1:4:0

Calculating

Fill with

- 1) Win
- 2) Lose
- 3) Tie

	Win	Lose	Tie
a	1	2	0
b	2	1	0
c	3	0	0
d	0	3	0

From 1st row
from

- 1) (a,b) → a loses
- 2) (a,c) → a loses
- 3) (a,d) → a winning

∴ 2 losses, 1 win

→ Candidate with highest wins will get highest ranking.
 \therefore c, b, a, d is the order.

- If any of the document got same wins, go for the doc with min^m losses.
- If same wins & same losses then go for random selection.

Q

$J_1 - P_1, P_2, P_3, P_4, P_5$
 $J_2 - P_2, P_1, P_4, P_3, P_5$
 $J_3 - P_3, P_4, P_5, P_2, P_1$
 $J_4 - P_4, P_3, P_2, P_1, P_5$

Pairs J_1 J_2 J_3 J_4

P_1, P_2	P_1	P_2	P_2	P_2
P_1, P_3	P_1	P_1	P_3	P_3
P_1, P_4	P_1	P_1	P_4	P_4
P_1, P_5	P_1	P_1	P_5	P_1
P_2, P_3	P_2	P_2	P_3	P_3
P_2, P_4	P_2	P_2	P_4	P_4
P_2, P_5	P_2	P_2	P_5	P_2
P_3, P_4	P_3	P_4	P_3	P_4
P_3, P_5	P_3	P_3	P_3	P_3
P_4, P_5	P_4	P_4	P_4	P_4

	P_1	P_2	P_3	P_4	P_5
P_1	-	1:3:0	2:2:0	2:2:0	3:1:0
P_2	3:1:0	-	2:2:0	2:2:0	3:1:0
P_3	2:2:0	2:2:0	-	2:2:0	4:0:0
P_4	2:2:0	2:2:0	2:2:0	-	4:0:0
P_5	1:3:0	1:3:0	0:4:0	0:4:0	-

	Win	Loss	Tie
P ₁	1	1	2
P ₂	2	0	2
P ₃	1	0	3
P ₄	1	0	3
P ₅	0	4	0

P₂ P₃ P₄ P₁ P₅

Ranking : OR

P₂ P₄ P₃ P₁ P₅

⇒ Linear Algebra.

$$Av = \lambda v$$

↓
Dataset

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

$$A = USV^T$$

Input →

Q $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ find eigen vectors & values

$$Av = \lambda v$$

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 2v_1 \\ 2v_2 \end{bmatrix}$$

$$2v_1 + v_2 = 2v_1$$

$$v_1 + 2v_2 = 2v_2$$

$$(2-2)v_1 + v_2 = 0$$

$$v_1 + (2-2)v_2 = 0$$

$$\therefore \begin{vmatrix} 2-2 & 1 \\ 1 & 2-2 \end{vmatrix} = 0$$

$$4 - 4\lambda + \lambda^2 - 1 = 0$$

$$\lambda^2 - 4\lambda + 3 = 0$$

$$(\lambda-1)(\lambda-3) = 0$$

$$\lambda_1 = 3, \lambda_2 = 1$$

$$\begin{aligned} -v_1 + v_2 &= 0 \\ v_1 + -v_2 &= 0 \end{aligned} \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{for } \lambda = 3$$

$$\begin{aligned} v_1 + v_2 &= 0 \\ v_1 + v_2 &= 0 \end{aligned} \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{for } \lambda = 1$$

⇒ Gram-Schmidt Normalization

→ Method of converting set of vectors into set of orthonormal vectors into a set of orthonormal vectors. It basically begins by normalizing the first vector under consideration & effectively rewriting the remaining vectors in terms of themselves - minus a multiplication of the already normalized vectors.

Q

$$A = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 \\ 1 & 2 & 1 & 7 \\ 0 & 2 & 0 \\ 2 & 3 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Q1ⁿ

$$\overrightarrow{V} \longrightarrow \overrightarrow{w} \longrightarrow \overrightarrow{U}$$

$$U_1 = \left\langle \frac{1}{\sqrt{6}}, 0, \frac{2}{\sqrt{6}}, \frac{1}{\sqrt{6}} \right\rangle$$



$$\overrightarrow{w}_k = \overrightarrow{v}_k - \sum_{i=1}^{k-1} \overrightarrow{v}_i \cdot \overrightarrow{v}_k * \overrightarrow{v}_i$$

$$\dots \quad \overrightarrow{w}_2 = \overrightarrow{v}_2 - \overrightarrow{v}_1 \cdot \overrightarrow{v}_2 * \overrightarrow{v}_1$$

$$= [2, 2, 3, 1] - \left[\frac{2}{\sqrt{6}} + 0 + \frac{6}{\sqrt{6}} + \frac{1}{\sqrt{6}} \right] * \left[\frac{1}{\sqrt{6}}, 0, \frac{2}{\sqrt{6}}, \frac{1}{\sqrt{6}} \right]$$

$$= [2, 2, 3, 1] - \frac{9}{\sqrt{6}} \left[\frac{1}{\sqrt{6}}, 0, \frac{2}{\sqrt{6}}, \frac{1}{\sqrt{6}} \right]$$

$$= [2, 2, 3, 1] - \left[\frac{3}{2}, 0, \frac{3}{2}, \frac{3}{2} \right]$$

$$\overrightarrow{w}_2 = \left[\frac{1}{2}, 2, 0, -\frac{1}{2} \right]$$

$$\overrightarrow{w}_3 = \overrightarrow{v}_3 - \overrightarrow{v}_1 \cdot \overrightarrow{v}_3 * \overrightarrow{v}_1 - \overrightarrow{v}_2 \cdot \overrightarrow{v}_3 * \overrightarrow{v}_2$$

Q

$$\overrightarrow{U_1} \quad \overrightarrow{U_2} \quad \overrightarrow{U_3}$$

$$\begin{bmatrix} \frac{1}{\sqrt{6}} & \cancel{\frac{\sqrt{2}}{6}} \\ 0 & \frac{2\sqrt{2}}{3} \\ \frac{2}{\sqrt{6}} & 0 \\ \frac{1}{\sqrt{6}} & -\frac{\sqrt{2}}{6} \end{bmatrix}$$

Sol^r

$$w_3 =$$

~~* singular~~

* Singular Value Decomposition.

- SVD is a method for transforming co-related variables into a set of uncorrelated variable that better expose the relationships among the original data.
- It is a method for identifying ordering the dimensions along which the data points exhibit the most variation.
- Therefore, it is possible to find the best approximation of the original points using fewer dimensions.
- Using SVD, a matrix A can be broken down into the product of 3 matrices. In orthogonal matrix U , a diagonal matrix S , & transpose of an orthogonal matrix V .

$$A = USV^T \quad , \text{ where } U^T U = I \quad \& \quad V^T V = I$$

↓ ↓
 orthogonal orthogonal

diagonal

- The columns of U are orthonormal eigen vectors of $A \cdot A^T$
- The columns of V are orthonormal eigen vectors of $A^T A$ & S is a diagonal matrix containing the sq. root of eigen values from U & V in descending order.

Q

$$A = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}, \text{ Apply SVD}$$

$$A_{2 \times 3} = U_{2 \times p} \underbrace{S}_{p \times p} V_{p \times 3}^T$$

As diagonal

→ The columns of U are orthonormal eigen vectors of $A \cdot A^T$

$$\text{Step 1: } A \cdot A^T = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix}$$

Now find eigen vectors of $A \cdot A^T$

$$(AA^T)v = 2v$$

$$\begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 2v_1 \\ 2v_2 \end{bmatrix}$$

$$11v_1 + v_2 = 2v_1$$

$$v_1 + 11v_2 = 2v_2$$

$$(11-2)v_1 + v_2 = 0$$

$$v_1 + (11-2)v_2 = 0$$

$$\begin{bmatrix} 11-2 & 1 \\ 1 & 11-2 \end{bmatrix} = 0$$

$$(11-2)^2 = 1$$

$$v_1 + v_2 = 0$$

$$v_1 - v_2 = 0$$

M	T	W	T	F	S	S
Page No.:	YOUVA					
Date:						

$$(11-2) \pm 1$$

$$\boxed{\lambda = 10, 12}$$

$$\cancel{\lambda = 10}$$

$$\cancel{v_1 + v_2 = 0}$$

Now assume $\lambda_1 = 12$.. so putting it in eqⁿ
 we get $v_1 = v_2$, we assume any point
 $[1, 1]$

→ Now at $\lambda_2 = 10$, so putting it we get
 $v_1 = -v_2$, we assume any point $[1, -1]$

Step 2: Eigen vectors

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \text{for second highest value of } \lambda$$

We have 2 conditions to check

- 1) Dot product 0 }
 - 2) Magnitude 1 }
- Any one satisfies will work.

$$v_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \therefore v_1 \cdot v_2 = 0$$

Condition satisfies

∴ Orthogonal vectors

∴ Now $v = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}$ (Converted to orthonormal form)

Step 3: Normalizing:

For matrix V

$$A^T A = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix}$$

Now eigen vector

$$(A^T A) \cdot V = 2V$$

$$\begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 2v_1 \\ 2v_2 \\ 2v_3 \end{bmatrix}$$

After solving we get $\lambda_1 = 12, \lambda_2 = 10, \lambda_3 = 0$

For $\lambda = 12$

$$\begin{aligned} 10v_1 + 2v_3 &= 12v_1 & v_1 = v_3 \\ 10v_1 + 4v_3 &= 12v_2 & 2v_3 = v_2 \\ 2v_1 + 4v_2 + 2v_3 &= 12v_3 & 2v_1 + 4v_2 = 10v_3 \therefore v_1 + 2v_2 = 5v_3 \end{aligned}$$

$\begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$ is
one of the solⁿ for $\lambda = 12$

For $\lambda = 10$

$$\begin{aligned} 10v_1 + 2v_3 &= 10v_1 & \therefore v_3 = 0 \\ 10v_2 + 4v_3 &= 10v_2 & \therefore v_3 = 0 \\ 2v_1 + 4v_2 &= 8v_3 & \therefore v_1 + 2v_2 = 4v_3 \therefore v_1 = -2v_2 \end{aligned}$$

$\therefore \begin{bmatrix} -2 & 1 & 0 \end{bmatrix}$ is
one of the solⁿ for $\lambda = 10$

For $\lambda = 0$

$$10v_1 + 2v_3 = 0 \quad \therefore v_3 = -5v_1$$

$$10v_2 + 4v_3 = 0 \quad 2v_3 = -5v_2$$

$$2v_1 + 4v_2 = 0 \quad \cancel{v_1 = -2v_2} \quad v_1 + v_3 = -2v_2$$

$+ 2v_3$

$\boxed{[1, 2, -5]}$
is one of the
sol" for $\lambda = 0$

$$\therefore \begin{bmatrix} 1 & -2 & 1 \\ 2 & 1 & 2 \\ 1 & 0 & -5 \end{bmatrix}$$

$$\begin{aligned} \vec{v}_1 &= (1, 2, 1) & v_1 \cdot v_3 &= 0 \\ \vec{v}_2 &= (-2, 1, 0) & v_2 \cdot v_3 &= 0 \\ \vec{v}_3 &= (1, 2, -5) & v_3 \cdot v_3 &= 0 \\ && \text{Check} & \text{part 5} \end{aligned}$$

$$\therefore V = \begin{bmatrix} \vec{v}_1 & \vec{v}_2 & \vec{v}_3 \end{bmatrix}$$

see calculate
the remaining
 $A = USV^T$

Topic 6.7.3 Latent Semantic (See from book)

⇒ Language Models.

1) Markov Models

Observable

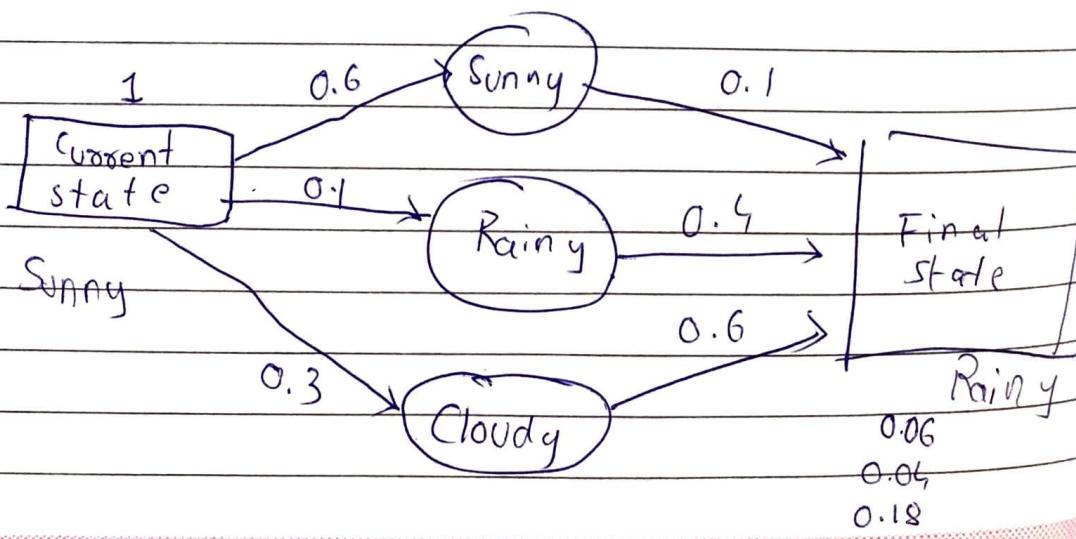
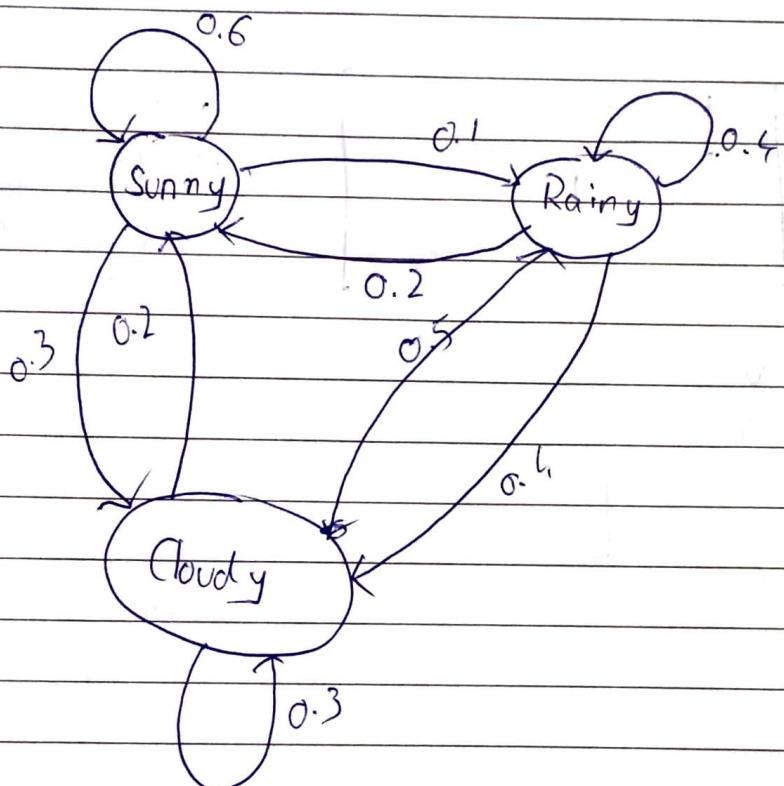
Hidden.

- Weather Prediction

- Sunny

- Rainy

- Cloudy



$$T = \begin{bmatrix} S & R & C \\ S & [0.6 & 0.1 & 0.3] \\ R & [0.2 & 0.6 & 0.4] \\ C & [0.2 & 0.5 & 0.3] \end{bmatrix}$$

state transition
prob. Matrix

$$P_0 = [P_1 \ P_2 \ P_3]$$

Initial state
Matrix.

(given in ques)

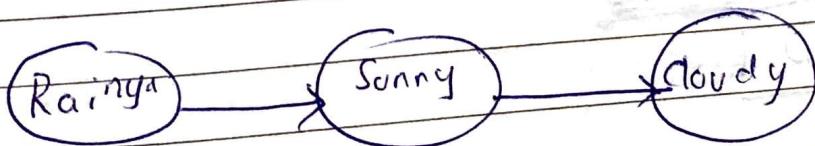
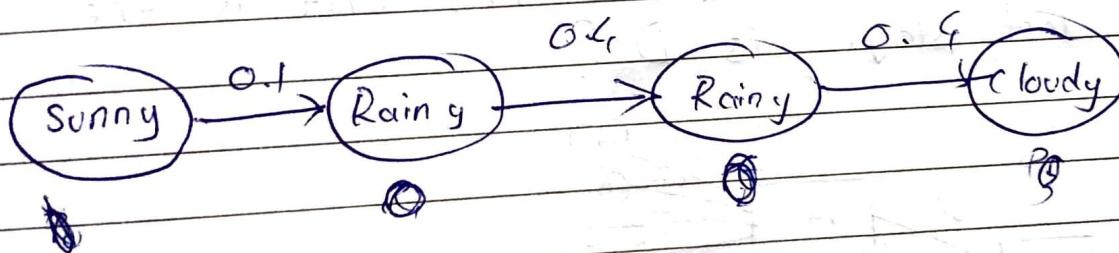
$$P_1 = P_0 \cdot T$$

$$P_2 = P_1 \cdot T$$

$$P_1 = P_0 T$$

$$P_2 = P_0 T^2$$

$$P_3 = P_0 T^3$$



Q

	A	B	C	D
A	0.1	0.2	0.3	0.4
B	0.5	0.1	0.3	0.1
C	0.4	0.2	0.1	0.3
D	0.3	0.1	0.3	0.3

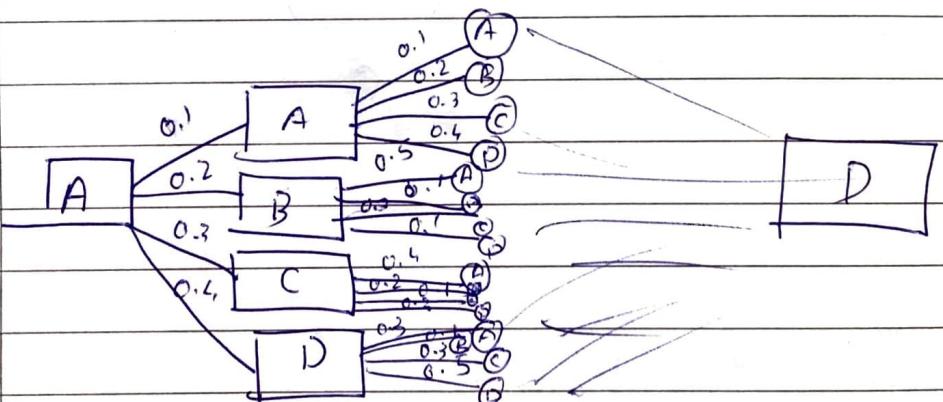
$T =$

→ To check whether ~~the sum of~~ state transition is valid or not, check sum of each row, it must be 1

$$P(ABCD) = P\left(\frac{B}{A}\right) P\left(\frac{C}{B}\right) P\left(\frac{D}{C}\right)$$

$$= 0.2 \times 0.3 \times 0.3 = 0.018$$

Initial state 1 is A. What is the prob that last stage is D?



$$\begin{aligned} P(AAAD) \\ P(ADDD) \end{aligned}$$

$$\text{Ans} = \sum_{i=A}^P \sum_{j=A}^D P(A_{ij}; D)$$

$$P_0 = [1 \ 0 \ 0 \ 0]$$

$$P_3 = P_0 T^3$$