Aayush : 🥲
Also Aayush : we don't do that here 🙌

OK PASS 69/69 ☺ Vijaya Ukraj

Page No.:
Date:

→ If recall increases, precision also increases
→ If precision " , recall " .

Eg.

		P(1)	R(1)
+	1	$\frac{1}{1}$ = 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| = { Jump, go, today, jumped, went, jumping }
|V| = { jump, go, today, went }

Query
|Q| = < Jumping >

D1 : Today went jumping
D2 : Go Jump today
D3 :- Jumped today

Alphabetical |V| = { go, jump, today, went }

B_1 = < 0, 1, 1, 1 > , Q = < 0, 1, 0
B_2 = < 1, 1, 1, 0 >
B_3 = < 0, 1, 1, 0 >

~~$B_3(1,1,1,0)$~~

~~$B = \frac{2}{\sqrt{3}\sqrt{3}} = \frac{2}{3}$~~

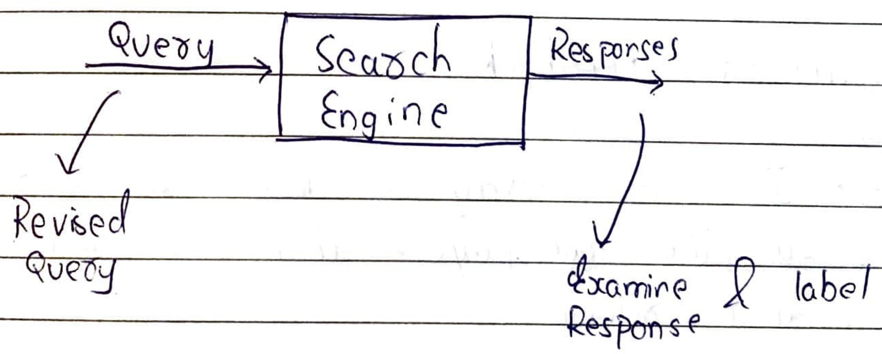
$$\cos(B_1, Q) = \frac{1}{\sqrt{3}}$$

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

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

⇒ Relevance feedback.

- Labelled learning



→ We ask user to label the responses as relevant or non-rel & then we will use labelled / supervised learning.

→ Revised query means for eg. if ~~5 wo~~ 5 query $k_1, k_2, \ldots, k_5$ & we found ~~k6~~ k3 is found in most non-relevant document, so we ~~will try~~ will revise our query by removing k3.

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

⇒ Methods

1) Rocchio method (Mentioned above adding or removing keywords)
2) Semi-supervised learning (Also known as ~~L-U~~ L-U learning) labelled-Unlabelled

2.1) L-U learning

2.2) P-U learning → Positive-Unlabelled

→ If user does not label the 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 ~~in~~ more active in the web-page, then we will ~~me~~ 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

Rank

Judges / Web Search engine

	J_1	J_2	J_3	J_4
P_1	1	2	5	4
P_2	2	1	4	3
P_3	3	4	1	2
P_4	4	3	2	1
P_5	5	5	3	5

3 methods

1) Broda (1770)
2) Reciprocal
3) Condorcet

1) Broda

Candidate - Voter method

Candidate → Participants

Voter → 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

Page No.:
Date:

$\therefore$ Candidate 1 = 5 + 4 + 1 + 2 = 12 points
" 2 = 4 + 5 + 2 + 3 = 14 "
" 3 = 3 + 2 + 5 + 4 = 14 "
" 4 = 2 + 3 + 4 + 5 = 14 "
" 5 = 1 + 1 + 3 + 1 = 6 "

→ 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.

	J1	J2	J3	J4
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
2 → Rank 4
1 → Rank 5

= 3

$\therefore$ Divided Equally $\therefore \frac{3}{2}$

$\therefore$ Cand 1 = 5 + 4 + 1.5 + 2
Cand 2 = 4 + 5 + 1.5 + 3
Cand 3 = 3 + 2 + 5 + 4

Same

M	T	W	T	F	S	S

Page No.:
Date:
YOUVA

2) Reciprocal

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

Borda Points distribution :- $n, n-1, n-2, \ldots 1$

Reciprocal " " :- $1, \frac{1}{2}, \frac{1}{3}, \ldots \frac{1}{n}$

→ For same ~~expa~~ example.

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

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

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

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

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

→ 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

For system 5, 'a' points

BORDA

$a = 4 + 3 + 2 + 1 + 1.5 = 11.5$ ✓ 3

$b = 3 + 4 + 3 + 3 + 3 = 16$ ✓ 1

$c = 2 + 1 + 4 + 4 + 4 = 15$ ✓ 2

$d = 1 + 2 + 1 + 2 + 1.5 = 7.5$ ✓ 4

Final Ranking :- B, C, A, D

RECIPROCAL

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

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

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

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

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.

Q ~~For~~ the same question

Soln

~~Pairwise~~

out of a & b, who has got highest rank for respective systems

Building NxN matrix

Pairs	S1	S2	S3	S4	S5
a,b	a	b	b	b	b
a,c	a	a	c	c	c
a,d	a	a	a	d	–
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
b	4:1:0	–	2:3:0	5:0:0
c	3:2:0	3:2:0	–	4:1:0
d	3:1:1	0:5:0	1:4:0	–

Win a
win b
Tie

Calculating
~~Fill with~~
1) Win
2) ~~lose~~
3) Tie

From 1st row
from
1) (a,b) → a losses
2) (a,c) → a losses
3) (a,d) → a winning
∴ 2 losses, 1 win

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

→ Candidate with highest ~~no~~ wins will get highest ranking. ∴ 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
T_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

Ranking :

$P_2 \quad P_3 \quad P_4 \quad P_1 \quad P_5$

OR

$P_2 \quad P_4 \quad P_3 \quad P_1 \quad P_5$

⇒ Linear Algebra.

$Av = \lambda v$, $\cancel{A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}}$

↓

Dataset

$A = USV^T$

Input ⟶ $\boxed{}$

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

$Av = \lambda v$

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

$$2v_1 + v_2 = \lambda v_1$$
$$v_1 + 2v_2 = \lambda v_2$$

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

$\therefore \quad \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} = 0$

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

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

$(\lambda-4)(\lambda-1) = 0$

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

$\left.\begin{matrix} -v_1 + v_2 = 0 \\ v_1 - v_2 = 0 \end{matrix}\right\}$ for $\lambda = 3$

$\left.\begin{matrix} v_1 + v_2 = 0 \\ v_1 + v_2 = 0 \end{matrix}\right\}$ for $\lambda = 1$

$\Rightarrow$ 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 & iteratively rewriting the remaining vectors in terms of themselves minus a multiplication of the already normalized vectors.

Q

$$A = \begin{array}{c} \begin{array}{ccc} \vec{v_1} & \vec{v_2} & \vec{v_3} \end{array} \\ \begin{bmatrix} 1 & 2 & 1 \\ 0 & 2 & 0 \\ 2 & 3 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

Sol^n

$$\vec{V} \longrightarrow \vec{W} \longrightarrow \vec{U}$$

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

$$\cancel{W_2 = \left\langle \frac{2}{\sqrt{18}}, \frac{2}{\sqrt{18}}, \frac{3}{\sqrt{18}}, \frac{1}{\sqrt{18}} \right\rangle}$$

$$\vec{W_k} = \vec{V_k} - \sum_{i=1}^{k-1} \vec{U_i} \cdot \vec{V_k} * \vec{U_i}$$

$$\therefore \vec{W_2} = \vec{V_2} - \vec{U_1} \cdot \vec{V_2} * \vec{U_i}$$

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

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

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

$$\vec{W_2} = \left[\frac{1}{2}, 2, 0, -\frac{1}{2}\right]$$

$$\vec{W_3} = \vec{V_3} - \vec{U_1} \cdot \vec{V_3} * \vec{U_1} - \vec{U_2} \cdot \vec{V_3} * \vec{U_2}$$

Q

$$\begin{array}{ccc} \vec{u}_1 & \vec{u}_2 & \vec{u}_3 \\ \left[\begin{array}{ccc} \frac{1}{\sqrt{6}} & \frac{\sqrt{2}}{6} & \\ 0 & \frac{2\sqrt{2}}{3} & \\ \frac{2}{\sqrt{6}} & 0 & \\ \frac{1}{\sqrt{6}} & -\frac{\sqrt{2}}{6} & \end{array}\right] & & \end{array}$$

Sol^n $\qquad w_3 =$

* ~~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.

$$\underbrace{U}_{\text{orthogonal}}\ \overbrace{S}^{\text{diagonal}}\ \underbrace{V^T}_{\text{orthogonal}}$$

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

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

→ The columns of V are orthonormal eigen vectors of A^TA & S is a diagonal matrix containing the sq. root of eigen values from U & V in descending order.

2 × 3

Page No.

Date

Q

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

As diagonal

$$A_{2\times3} = U_{2\times p} \; S_{p\times p} \; V^T_{p\times3}$$

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

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

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

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

$$(AA^T)v = \lambda v$$

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

$$11v_1 + v_2 = \lambda v_1$$
$$v_1 + 11v_2 = \lambda v_2$$

$$(11-\lambda)v_1 + v_2 = 0$$
$$v_1 + (11-\lambda)v_2 = 0$$

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

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

$v_1 + v_2 = 0$
$v_1 - v_2 = 0$

$(11 - \lambda) = \pm 1$

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

~~At $\lambda = 10$~~

~~$v_1 + v_2 = 0$~~

Now ~~put~~ assume $\lambda_1 = 12$ $\therefore$ so putting it in eq^n 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 $[1, -1]$

$$\therefore \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

for second highest value of λ

column corresponding to highest λ value

Step 2 :- Eigen vectors

We have 2 conditions to check
1) Dot product 0
2) Magnitude - 1
} Any one satisfies ~~will work.~~

$v_1 = [1, 1]$
$v_2 = [1, -1]$

$\therefore v_1 \cdot v_2 = 0$
Condition 1 satisfies.
$\therefore$ Orthonormal vectors.

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

Step 3 :- Normalizing.

Date:

For matrix V

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

Now eigen vector

$$(A^TA) \cdot v = \lambda v$$

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

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

For $\lambda = 12$

$10v_1 + 0 + 2v_3 = 12v_1$ $\qquad v_1 = v_3$

$0 + 10v_2 + 4v_3 = 12v_2$ $\qquad 2v_3 = v_2$

$2v_1 + 4v_2 + 2v_3 = 12v_3$ $\qquad 2v_1 + 4v_2 = 10v_3 \therefore v_1 + 2v_2$

$[1, 2, 1]$ is one of the sol^n for $\lambda = 12$

For $\lambda = 10$

$10v_1 + 2v_3 = 10v_1$ $\qquad \therefore v_3 = 0$

$10v_2 + 4v_3 = 10v_2$ $\qquad \therefore v_3 = 0$

$2v_1 + 4v_2 = 8v_3$ $\qquad \therefore v_1 + 2v_2 = 4v_3$ $\qquad \therefore v_1 = -2v_2$

$\therefore [-2, 1, 0]$ is one of the sol^n for $\lambda = 10$

For $\lambda = 0$

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

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

$2v_1 + 4v_2 + 2v_3 = 0$ ~~$v_1 = 2v_2$~~ $v_1 + v_3 = -2v_2$

[redacted] $\begin{bmatrix} 1, 2, -5 \end{bmatrix}$ is one of the sol^n for $\lambda = 0$

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

$\vec{v}_1 = (1, 2, 1)$
$\vec{v}_2 = (-2, 1, 0)$
$\vec{v}_3 = (1, 2, -5)$

$v_1 \cdot v_3 = 0$
$v_1 \cdot v_3 = 0$
$v_2 \cdot v_3 = 0$
Check pairwise

$\therefore$ $V = [$

Calculate the remaining
$A = USV^T$

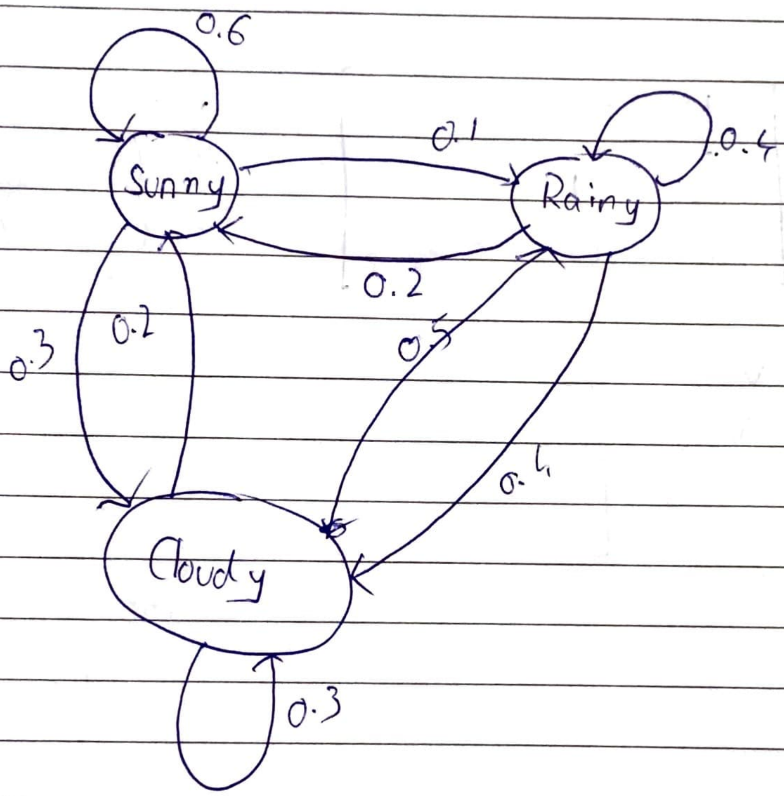
(See from book)

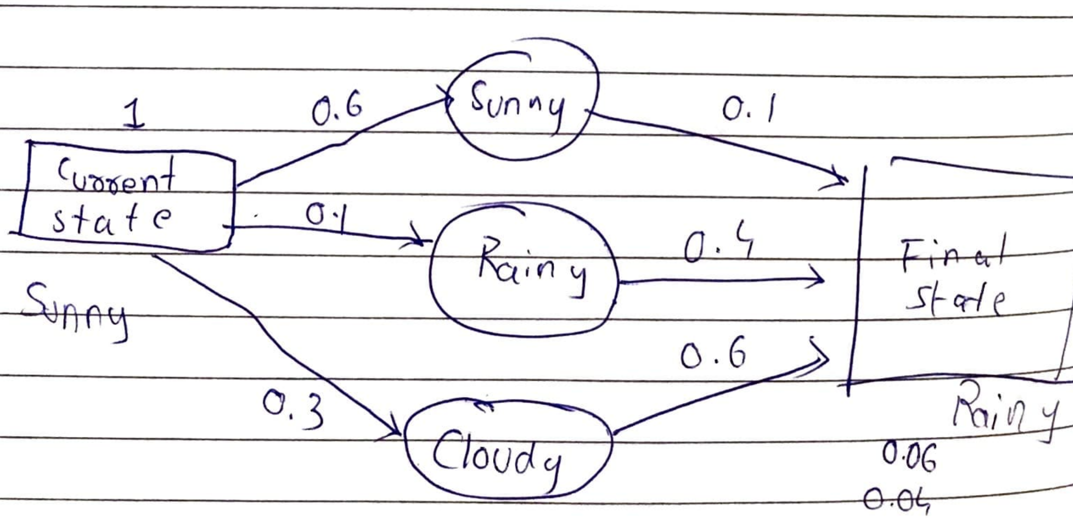
Topic 6.2.3 Latent Semantic

⟹ Language Models.

1) Markov Models — Observable / Hidden.

- Weather Prediction
  - Sunny
  - Rainy
  - Cloudy





0.06
~~0.04~~
0.18
0.25

Sunny

$$T = \begin{array}{c|c} & \begin{array}{ccc} S & R & C \end{array} \\ \begin{array}{c} S \\ R \\ C \end{array} & \begin{bmatrix} 0.6 & 0.1 & 0.3 \\ 0.2 & 0.6 & 0.4 \\ 0.2 & 0.5 & 0.3 \end{bmatrix} \end{array} \longleftarrow$$ state transition prob. Matrix

$$P_0 = \begin{bmatrix} P_1 & P_2 & P_3 \end{bmatrix} \longleftarrow$$ Initial state Matrix.

(given in ques)

$P_1 = P_0 \cdot T$

$P_2 = P_1 \cdot T$

$P_1 = P_0 T$

$P_2 = P_0 T^2$

$P_3 = P_0 T^3$

P_1

$$\text{Sunny} \xrightarrow{0.1} \text{Rainy} \xrightarrow{0.4} \text{Rainy} \xrightarrow{0.4} \text{Cloudy}$$

P_3

$$\text{Rainy} \longrightarrow \text{Sunny} \longrightarrow \text{Cloudy}$$

$0.3 \times 0.2 \times 0.4$

Q

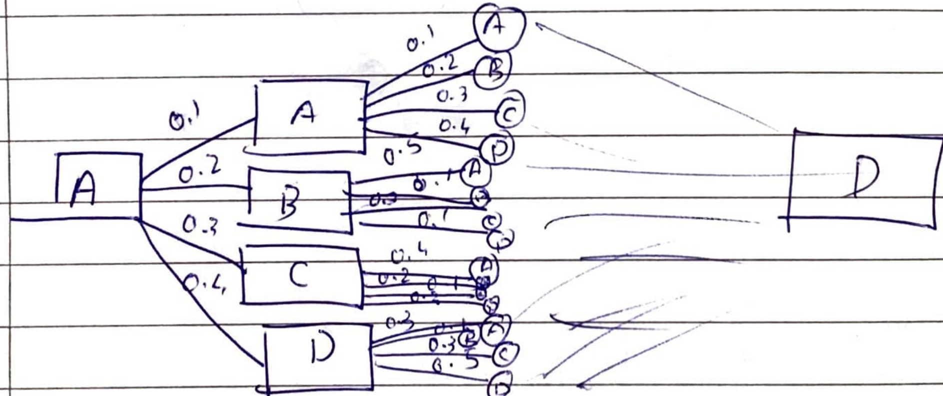
$$T = \begin{array}{c|cccc} & A & B & C & D \\ \hline A & 0.1 & 0.2 & 0.3 & 0.4 \\ B & 0.5 & 0.1 & 0.3 & 0.1 \\ C & 0.4 & 0.2 & 0.1 & 0.3 \\ D & 0.3 & 0.1 & 0.3 & 0.3 \end{array}$$

→ To check whether ~~so 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?



$P(AAAD)$
$P(ADDD)$

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

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

$$P_3 = P_0 T^3$$